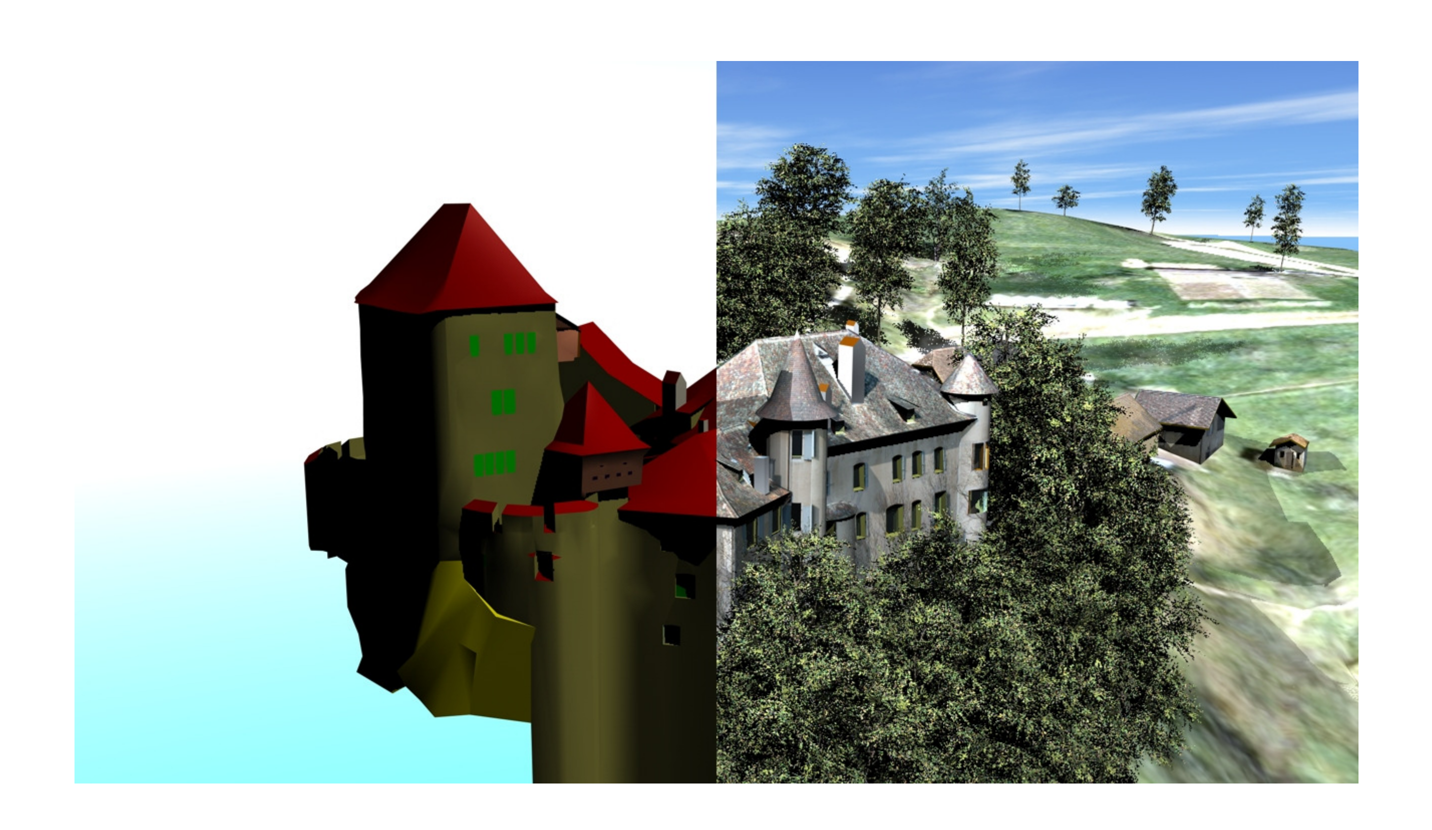
Fachhochschule
beider Basel
Nordwestschweiz
FHBB

Departement
Bau

Vermessung
und Geoinformation

Diplomarbeit 2005/11

Texturierung, Visualisierung und Animation komplexer 3D-Geoobjekte



Diplomanden: Thomas Schuler
David Emmenegger

Examinator: Prof. Dr. S. Nebiker

Experte: K. Budmiger

Datum: 22. November 2005

Vorwort

Sinn und Zweck einer Diplomarbeit ist unter anderem, dass sich die Studenten während dem Zeitraum von 10 Wochen intensiv in einem Gebiet vertiefen können. In dieser Arbeit kann der über 6 Semester gelernte Stoff angewendet und im jeweiligen Gebiet vertieft werden.

Es sollen dabei auch die vier Hauptkompetenzen, Sozial-, Fach-, Methodik- und Selbstkompetenz gefördert werden. Diese Arbeit soll den abschliessenden Beweis der Reife in allen Kompetenzen erbringen

Im Zuge der Sozialkompetenz möchten wir allen Personen, welche uns in irgendeiner Form unterstützt haben, herzlich danken. Sei dies in der Funktion als Lektor, welche den Bericht durchgelesen haben, oder ganzen Schulklassen, welche uns ihr Klassenzimmer mitsamt ihrer Kaffeemaschinen zur Verfügung stellten. Allen anderen Personen, die uns unterstützt haben und nicht erwähnt wurden gebührt unser Dank auch.

Abstract

Die attraktive Präsentation von 3D-Geoobjekten wird immer wichtiger. Es wird immer häufiger versucht, realitätsnahe Abbildungen zu erzeugen. Deshalb werden die Geoobjekte immer häufiger mit Foto-Texturen versehen, oder sogar animiert. Wichtige Forschungspunkte sind daher die Darstellung von Vegetation und die Erzeugung von realitätsnahen Objekten.
Ziel dieser Diplomarbeit ist es, Möglichkeiten aufzuzeigen wie komplexe 3D-Geoobjekte dargestellt und präsentiert werden können.
Die kommerzielle 3D-Modellierungssoftware Cinema 4D von Maxon steht uns als Arbeitsmittel für diese Aufgabe zur Verfügung. Bei den Untersuchungen wird eine Unterteilung in spezifische Fachgebiete vorgenommen.
Für die Texturierung wurde bis anhin an der FHBB die Software DILAS verwendet. Die Texturierung erfolgte für jedes einzelne Polygon, was bei grossen Objekten viel Zeit beansprucht. In dieser Arbeit wird dargelegt, dass eine einfache Texturierung der 3D-Geoobjekte nun möglich ist. Es müssen nicht mehr alle Polygone einzeln texturiert werden. Die Präsentation in anderen Medien in Echtzeitdarstellung wird über einen Export in das VRML-Format auch ermöglicht.
Um eine möglichst realitätsgetreue Visualisierung der Realwelt zu erhalten, ist es unverzichtbar, auch die Vegetation mit einzubeziehen. Dies wird mit der Plug-In-Vegetationsbibliothek von Cinema 4D realisiert. Weiter ist von Interesse, wie aus terrestrischen Laserscanningdaten Bäume modelliert werden können. Aufgrund der erfassten Laserscanningdaten können die Bäume an exakte Positionen gesetzt und in ihrer Form angepasst werden. Untersuchte Methoden zur Modellierung von Bäumen aus Laserscanningdaten führten zu keiner definitiven Lösung. Die Darstellung einer Wiese ist schwierig, die Verwendung eines guten Orthophotos hat sich als genügend erwiesen.
Um die erstellten Abbildungen einem breiten Publikum attraktiv zu zeigen, wird die Wirkungskraft eines kleinen Werbefilms ausgenützt. Leider können die Cinema 4D-Szenen nicht in voller Qualität in Echtzeit dargestellt werden. Darum ist eine Präsentation mittels Filmdarstellung notwendig. Dabei spielt ein speziell erstelltes Drehbuch für das gewählte Zielpublikum eine zentrale Rolle. Dies wird in praktischen Beispielen umgesetzt. Bei der Planung der bisherigen Projekte wurde auf eine Dokumentation des Arbeitsablaufes verzichtet. Es wurde ein optimaler zukünftiger Ablauf von der Planung bis zur Animation für die Bereitstellung von kulturellen wertvollen Objekten erstellt.
Eine Echtzeitvisualisierung der vollständig erstellten Szene mit Vegetation wird mit der Software "G-Vista Suite 2005" ermöglicht. Dabei konnte nicht die volle Qualität der Objekte aus Cinema 4D übernommen werden. Die Übernahme der 3D-Objekte aus Cinema 4D in die Software G-Vista ist mit VRML (Virtual Reality Modelling Language) möglich.
Es hat sich gezeigt, dass realitätsnahe Visualisierungen mit Cinema 4D möglich sind. Sie sind aber immer noch mit hohem Zeitaufwand verbunden und könnten noch besser automatisiert werden. Trotz Verwendung von Vegetationsbibliotheken mit wenigen Polygonen explodiert die Datenmenge bei Verwendung vieler Vegetationsobjekte.

Redlichkeit

Ich versichere, dass ich die vorliegende Arbeit selbstständig und ohne Verwendung anderer als der im Quellenverzeichnis (Literaturverzeichnis) angegebenen Quellen und Hilfsmittel angefertigt habe. Die wörtlich oder inhaltlich der im Literaturverzeichnis aufgeführten Quellen und Hilfsmittel entnommenen Stellen sind in der Arbeit als Zitate kenntlich gemacht und mit einem Verweis auf das Literaturverzeichnis versehen. Diese Diplomarbeit ist nicht veröffentlicht, keinen anderen Interessenten zugänglich gemacht und keiner anderen Prüfungsbehörde vorgelegt worden.

Muttenz, 22. November 2005

David Emmenegger Thomas Schuler

1 Einleitung

Verfasser: Th. Schuler und D. Emmenegger

Die attraktive Präsentation von 3D-Geoobjekten wird immer wichtiger. Auch werden die Geoobjekte immer häufiger mit Texturen versehen oder sogar animiert. Es wird öfters versucht, realitätsnahe Abbildungen zu erzeugen. Zu den Momentan wichtigen Forschungspunkten gehören die Darstellung von Vegetation und die Erzeugung von realitätsnahen texturierten Objekten.
Ziel dieser Diplomarbeit ist es, Aussagen zu machen über die Möglichkeiten der detailgetreuen Texturierung und Präsentationen von komplexen 3D-Geoobjekten.

1.1 Motivation

In den letzten Jahren wurde an der FHBB mit dem Projekt "3D Cultural Heritage" viel in der effizienten Modellierung komplexer 3D-Geoobjekte unternommen. Mittlerweile stellt die messtechnische Erfassung von grossen Geoobjekten (z.B. Schlösser) kein eigentliches Problem mehr dar. Auch konnte eine Grundlage für die Speicherung der Daten mit DILAS geschaffen werden. Die heutigen Schwierigkeiten liegen nun bei der Datenaufbereitung, bei einem optimalen einfachen Datenfluss ohne Überschneidungen und bei der Präsentation der Daten.

1.2 Ausgangslage

Aus diesem Problemkreis folgte unsere Aufgabenstellung für diese Diplomarbeit. Die Aufgabenstellung gliedert sich in folgende Punkte:

Cinema 4D

Untersuchung der Funktionalität des Softwaretools Cinema 4D zur Modellierung, Texturierung, Visualisierung und Animation komplexer 3D-Geoobjekte. Sowie einer Dokumentation der typischen Arbeitsabläufe in Cinema4D.

Texturierung

Die abstrahierte und photorealistische Texturierung komplexer 3D-Objekte ist zu untersuchen. Welche Möglichkeiten zur Texturierung stellen die unterschiedlichen 3D-Tools und Formate zur Verfügung (z.B. VRML, Microstation, Cinema 4D) oder wie lassen sich unterschiedliche Detaillierungsstufen für Geometrie und Texturen erzeugen und handhaben?

Vegetation

Untersuchung unterschiedlicher Möglichkeiten und Werkzeuge zur Modellierung und Visualisierung von Vegetation und Erstellung einer fundierten Zusammenstellung, inkl. Beurteilung.

Visualisierung im Internet

Visualisierung komplexer 3D-Objekte im Internet. Dazu sollen einerseits die Möglichkeiten zur Vereinfachung von Geometrie und Texturen untersucht werden. Andererseits sollen die Möglichkeiten zur Übernahme von 3D-Modellen aus Cinema 4D in die Software G-Vista Suite 2005 der Firma Geonova AG untersucht und aufgezeigt werden.

Videothematik

Untersuchung und Aufzeigen aktueller Möglichkeiten zur attraktiven Präsentation mittels Videoanimationen.

Optimaler Datenfluss

Erarbeitung und Dokumentation eines optimalen Datenflusses für die zukünftige Modellierung, Texturierung und Visualisierung komplexer 3D-Geoobjekte (z.B. für zukünftige Projekte im Bereich '3D Cultural Heritage')

Vergleich Microstation vs. Cinema 4D

Erstellen eines fundierten Vergleichs von Funktionalität und Bedienungsanforderungen zwischen Microstation und Cinema 4D.

2 Cinema 4D

Verfasser: Th. Schuler (2.6-2.8) und D. Emmenegger (2.1-2.5)

Bei Cinema 4D handelt es sich um ein professionelles 3D-Animierungstool. Es können damit 3D-Szenen erstellt und anschliessend als Einzelbilder ausgegeben werden. Im folgenden werden einige wichtige Funktionen näher erklärt. Es ist nicht unsere Absicht, alle Funktionen des Programms zu erläutern. Vielmehr sollen die Funktionen, welche zur Erstellung einer 3D-Szene wichtig sind, erklärt werden. Für die restlichen Funktionen verweisen wir gerne auf das Handbuch und die Online-Hilfe.

2.1 Export/Importmöglichkeiten

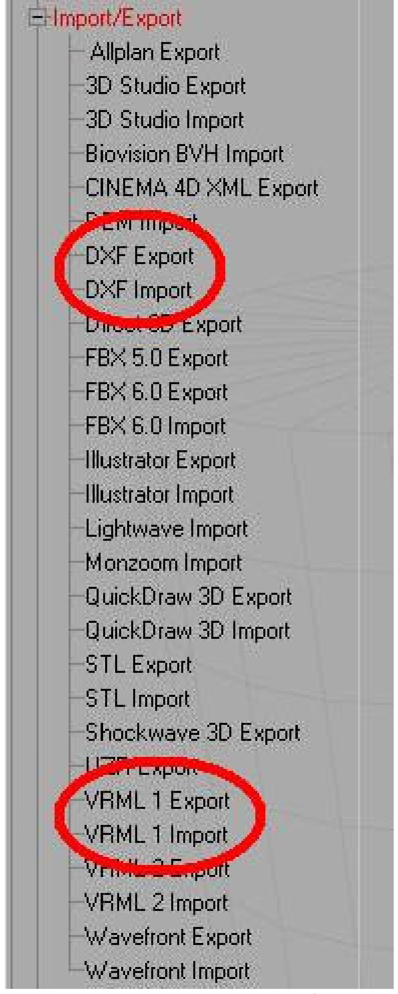


Abbildung 1 Import/Export in Cinema 4D

Cinema 4D bietet eine grosse Vielfalt von Import- und Exportmöglichkeiten. Bei den getätigten Untersuchungen beschränkten wir uns auf die Formate, welche bei der täglichen Arbeit am meisten zum Zug kommen. Diese ist einerseits das DXF-Format, welches die Schnittstelle zu den CAD's sicherstellt und andererseits das VRML-Format welches bevorzugt im Bereich der Internetpräsentation verwendet wird.

2.1.1 DXF

Die Tests haben ergeben, dass sich das DXF-Format gut für den Austausch zwischen den CAD-Programmen und Cinema 4D eignet. Der Import von DXF Dateien gestaltet sich relativ einfach.

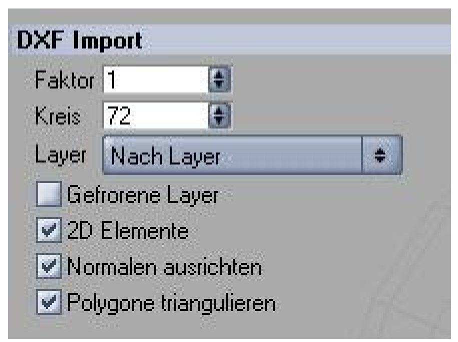


Abbildung 2 DXF Importoptionen

Faktor
Dies ist der Skalierungsfaktor, welcher beim Importieren des DXF angewendet wird.
Kreis
Hier wir die Anzahl Segmente angegeben, in welcher ein Kreis beim Import aufgeteilt wird.
Layer
Ein DXF-File besteht meist aus vielen einzelnen Elementen. Hier besteht eine Möglichkeit, diese Elemente nach gewissen Optionen zusammenzufügen.

- nach Farbe
- nach Layern
- nicht Verbinden (was gar nichts macht)

Gefrorener Layer
Gibt an, ob ausgeblendete Layer (gefrorene) in sichtbare konvertiert werden sollen oder nicht.
2D-Elemente
Sagt, ob 2D-Elemente überhaupt eingelesen werden sollen.
Normalen ausrichten
Dabei können die Flächennormalen des DXF-Files einheitlich ausgerichtet werden.
Polygone triangulieren
Aus dreidimensionalen Polygonzügen wird beim Import eine Fläche erzeugt.

Die besten Importergebnisse erzielten wir mit den Einstellungen, welche in Abbildung 2 ersichtlich sind. Da die Einstellungsmöglichkeiten bei Cinema 4D relativ rudimentär sind, ist den Exportparametern im CAD sehr viel Beachtung zu schenken.

Bei DXF-Files besteht meist noch das Problem, in der Version, in der sie gespeichert sind. Cinema 4D kann auch nicht alle Versionen importieren. Es werden nur die folgenden Versionen unterstützt:

- AutoCAD Dateien bis einschliesslich Version 12
- **100%** kompatiblen Exportfiltern

Weiter müssen folgende Einstellungen in Microstation vorgenommen werden, um einen erfolgreichen Export zu ermöglichen.

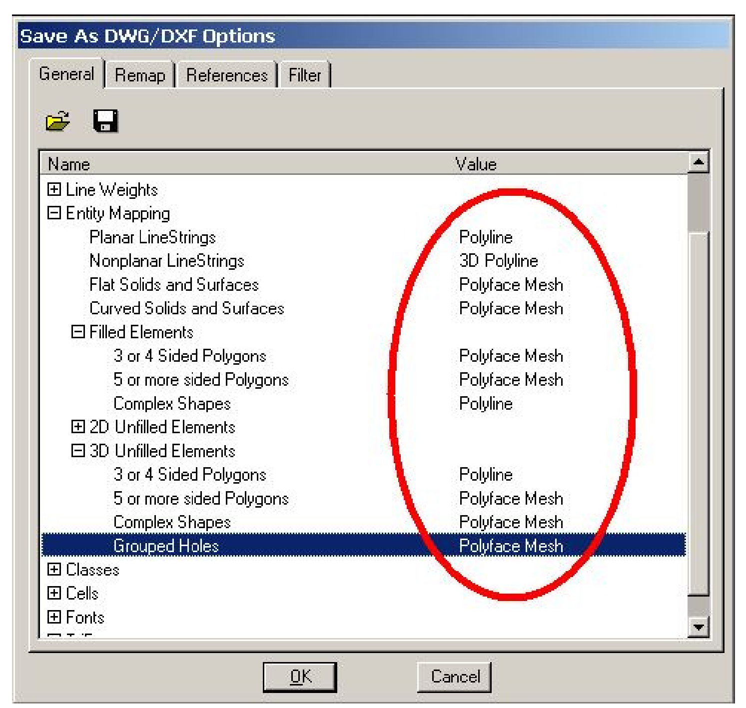


Abbildung 3 Exporteinstellungen aus Microstation

Bei den Versuchen hat sich gezeigt, dass je nach verwendeten Grundlagen die Einstellungen sich verändern können. Leider kann dieses Problem nicht gelöst werden und es müssen bei jedem Export die geeigneten Einstellungen gefunden werden.

2.1.2 VRML

Um ein einzelnes Objekt auch im Web präsentieren zu können ist das VRML-Format gut geeignet. Hierbei soll nicht die gesamte Szene exportiert werden, sondern nur die Hauptobjekte. Das Höhenmodell sowie das Orthophoto werden durch den Internetviewer (in diesem Fall G-Vista) bereitgestellt.



Abbildung 4 VRML Exportoptionen

Faktor
Dieser stellt den Skalierungsfaktor dar, welcher beim VRML Export angewendet wird.
Text formatieren
Mit dieser Option wird das exportierte VRML-File so formatiert, dass ein manuelles Bearbeiten einfacher ist.
Backface-Culling
Hiermit wird das Zeichnen der Objektrückseiten ausgeschaltet. Die Darstellung wird dadurch wesentlich beschleunigt. Wird durch G-Vista aber nicht unterstützt.
Texturen
In VRML können die Texturen entweder integriert oder als einzelne Dateien angehängt werden.

Um einen VRML-Export zu ermöglichen, müssen alle Texturen UVW-Koordinaten zugewiesen werden. Das heisst, die Texturen werden auf der Oberfläche des Objektes fixiert. Dies geschieht im Objektmanager unter dem Menü Textur. Die Textur-Kameras werden anschliessend nicht mehr gebraucht, die Texturen sind jetzt auf dem Objekt fixiert.

2.2 Unterstützte Koordinatenwerte

Bei den gemachten Untersuchungen stellte sich schnell ein Problem bei der Bearbeitung in Cinema 4D ein. Es war relativ schwierig, in der Szene zu navigieren. Vor allem das Zoomen „hackte" sehr stark, was zu grossen Zoomsprüngen führte. Es traten noch weitere Probleme auf:

- Navigieren (Zoomen, Drehen, Schieben)
- Renderungsungenauigkeiten
- Finden der importierten Szene

Die Ursache dieser Probleme liegt darin, dass Cinema 4D die Szene im Koordinatenursprung erwartet. Programm intern wird ohne Kommastellen, auf eine fixe Anzahl Stellen gerechnet. Koordinatenwerte wie z.B. 600'000.000 oder 200'000.000 liegen anscheinend ausserhalb dieser Anzahl Stellen, was zu Rechenungenauigkeiten führt.

Die Lösung dieses Problems ist, die gesamte Szene mit einem Shift in die Nähe des Nullpunktes zu verschieben. Vorzugsweise geschieht dies vor dem Export aus dem CAD. So entstehen keine Probleme beim Import in Cinema 4D und der Navigation.

2.3 Winkelsystem

Cinema 4D verwendet nicht das Winkelsystem, welches in der Photogrammetrie üblich ist. Wie aus der Abbildung 5 ersichtlich ist, werden die Winkel mit **H**eading, **P**itch und **B**ank benannt.



Abbildung 5 Koordinatenwerte in Cinema 4D

Dabei wird das Objekt nicht um die eigenen Objektachsen gedreht, sondern um die Achsen des übergeordneten Winkelsystems. Der Grund, wieso man nicht einfach die Drehungen um die eigenen Objektachsen nimmt, liegt in der Mathematik. Bei Drehungen um die Objektachsen bilden die Winkel eine nicht kommutative Gruppe. Die Reihenfolge der Winkel ist dabei entscheidend. Für Animationen ist aber dieses System nicht brauchbar. Bei dem HPB-System ist die Reihenfolge der Winkel unwichtig. Welche Reihenfolge man auch wählt, man kommt immer zum gleichen Ergebnis.

2.4 Net Renderer

Mit dem Net Renderer steht ein sehr nützliches Tool zum gleichzeitigen renderen auf mehreren Computern zur Verfügung. Die Voraussetzungen, um einen Computer zum renderen über das Netzwerk einzusetzen sind:

- Net Renderer Client muss installiert sein
- Computer müssen am gleichen Netzwerk sein
- Clients sollten während des Renderen nicht benützt werden

Die Daten werden während des Renderen auf die einzelnen Computer aufgeteilt. Diese Aufteilung wird dynamisch nachgeführt. Mit anderen Worten: Sobald ein Computer mit seinen Berechnungen fertig ist wird geprüft, ob es noch zu berechnende Daten gibt. Diese Daten werden, sofern Rechenkapazität frei ist, verteilt. Somit wird eine optimale Auslastung der Hardware gewährleistet.

2.5 Tauglichkeit für die Videoerstellung

Grundsätzlich wäre es möglich, ganze Filme in Cinema 4D zu erzeugen. Allerdings ist dies relativ umständlich. Es werden im Moment nur unkomprimierte WAV-Musikfiles unterstützt, was die Filme unnötig gross macht. Auch ist die Wahl der Komprimierungsverfahren sehr eingeschränkt. Diese Verfahren sind gerade für den Austausch via Internet wichtig, da mit ihnen die Datengrösse stark verringert werden kann.
Gerade wenn man einen längere Szene rendern will, ist die Möglichkeit eines Systemabsturzes vorhanden. Wenn diese Szene nun als Film in Cinema 4D erstellt wird, gehen bei einem Absturz alle Daten verloren. Werden aber Einzelbilder gerendert, bleiben die schon gerenderten Bilder erhalten. Der Datenverlust wird somit minimiert.

Aus diesen Gründen ist es empfehlenswert, in Cinema 4D nur Einzelbilder zu renderen und diese anschliessend in einem Filmbearbeitungsprogramm zu einer Szene zusammenzufügen. Es können in diesem Programm mehrere Szenen zusammengefügt und vertont werden. Es sind auch Spezialeffekte wie Überblendungen oder Standbilder möglich.

2.6 Messen + Konstruieren

Das „Messen & Konstruieren“-Werkzeug, welches unter der Befehlsleiste „Funktionen“ zu finden ist, wird in dieser Arbeit vor allem zur Kontrolle und nachträglicher Änderung der Form und Höhe von Bäumen verwendet. Dabei wird das zu messende Objekt im Objekt-Manager aktiv gesetzt. Nun kann unter dem gleichzeitigen drücken der CTRL- und SHIFT-Taste der Startpunkt angewählt werden. Der Endpunkt wird mittels drücken der SHIFT-Taste gesetzt. Die gemessene Distanz steht nun einerseits im Bild selber, gekennzeichnet mit einem roten Pfeil und im Attribute-Manager. Beim skalieren des selektierten Objekts wird die Bemassung laufend angepasst. Somit kann ein Objekt exakt auf eine gegebene Form geändert werden.

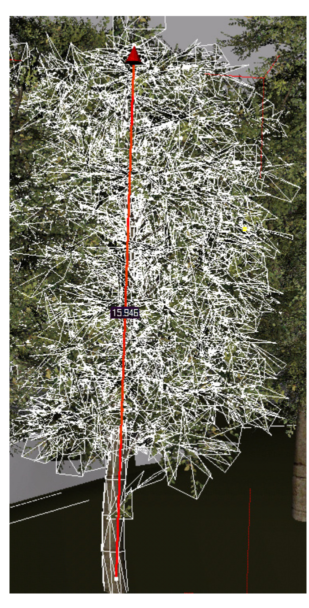


Abbildung 6 Beispiel einer Messung

2.7 Polygonreduktion

Oft sind durch die automatische Dreiecksvermaschung eines verwendeten Programms viele Polygone zu beklagen, was bis anhin für Texturierungsaufgaben nicht hilfreich war, da in DILAS jedes Polygon für sich texturiert wurde. Aber auch im Bezug auf die Datenmenge sind zu viele Polygone nicht hilfreich, solange die Geometrie eines Objektes den projekt-spezifischen Genauigkeitsanforderungen noch genügend ist. Es stellt also so oder so eine wichtigen Aspekt dar, wie man die Anzahl Polygone einfacher reduzieren kann.

In Cinema 4D kann unter den Deformations-Objekten eine Polygonreduktion aufgerufen und einem Polygon-Objekt zugewiesen werden. Die beste und schnellste Art, die Polygonanzahl zu verringern, ist über den Prozentschieber. Die Verringerung der Polygonanzahl wird unter gleichzeitiger möglichst weitgehender Erhaltung der geometrischen Objektform durchgeführt. Es gibt zusätzlich noch Optionen, um den Kompromiss zwischen Rechengeschwindigkeit und Genauigkeit genauer zu bestimmen. Der Mesh-Qualitätsfaktor überprüft, ob es überlappende Polygone gibt und gegebenenfalls werden diese in Abhängigkeit des Fakotrs beseitigt. Eine weitere wichtige Option ist die Grenzkanten-Erhaltung. Bei aktivem Button werden die „Grenzen“ eines nicht geschlossenen Objektes erhalten bleiben.

Das Resultat kann graphisch verfolgt und entsprechend den Wünschen angepasst werden. Das Gute dabei ist, dass die Möglichkeit besteht einzelne Objekte unterschiedlich stark zu reduzieren. Gerade die Option der Grenzkanten-Erhaltung ist ein grosser Vorteil gegenüber einer Dreiecksverdünnung in Cyclone.

2.8 Animationen / Darstellung

Objekte wie Rauch / Feuer / Springbrunnen werden alle mit Hilfe von „Partikelsysteme“ / „Thinking Particels“ / „XPresso“-Schaltungen und Shaders erstellt.

Partikelsysteme

Grundlage des Partikelsystems ist ein Emitter, der einen Strom von Teilchen (Partikel) ausspuckt. Zahlreiche Parameter und Modifikatoren steuern das zeitliche und räumliche Verhalten der Partikel. (Manuel_C4D S. 321ff.) [6]

XPresso

Mit dem Node-Editor „XPresso“ können komplexe, automatisierte Objektinteraktionen relativ leicht bewerkstelligt werden. Solche Abhängigkeiten baut man im XPresso-Editor zusammen, welcher sich im Objektmanager unter Datei/Cinema4d Tags/XPesso findet lässt. (Manuel_C4D S.893ff.)[6]

Thinking Particels

“Thinking Particels” ist ein überaus komplexes Partikel-System, das über den XPresso-Editor und den Attribute-Manager gesteuert wird. Es unterscheidet sich dabei grundlegend vom Standard-Partikel-System von Cinema 4D. Mit „Thinking Partikels“ erhält man Kontrolle über jeden Aspekt eines Partikelstroms. Aufgrund der Fülle von Kontrollmöglichkeiten und Parametern sollte „Thinking“, also das „Nachdenken“, durchaus wörtlich genommen werden. (eigenes Manuel)[7]

Shader

Shader sind mathematisch berechnete Texturen, die zwei oder dreidimensional sein können. Der Vorteil der Shader ist, dass man unendlich nahe heranzoomen kann, ohne jemals ein Pixel zu sehen. Shader erzeugen Oberflächen, die mit Bitmap-Texturen oder anderen herkömmlichen Methoden nicht zu realisieren. Alle Shader können direkt im Material-Manager über das Menü „Datei / Shader“ aufgerufen werden. (Manuel_C4D S. 721ff.)[6]

2.8.1 Springbrunnen

Als Grundlage dient ein „P Sturm“ also „Thinking Particels“ welcher vergleichbar ist mit einem normalen Emitter der vorgegebenen Partikelsysteme. Er wird knapp unterhalb der Wasserfläche platziert. Natürlich muss der Emitter so ausgerichtet werden, dass die entstehenden Partikel senkrecht von der Wasserfläche weg strömen. Als Partikelform dient eine einkleinwenig abgeänderte Kapsel, welche in genügender Anzahl das Erscheinungsbild eines Wasserstrahls erhält. Es werden „Thinking Particels“ verwendet um die Ausrichtung der Kapseln zu manipulieren.
Um die ausgestossenen Partikel wieder in Richtung ihres Ursprungs zu zwingen, wird ein „P Gravitaion“ eingefügt. Mit diesem Modifikator kann der Effekt eines Gravitationsfeldes nachgebildet werden.
Um Wellen darzustellen wird der Wasserfläche ein Wellen-Shader zugewiesen. Er lässt dort, wo Partikel auf ein Objekt treffen, wellenförmige, animierte Graustufen-Effekte entstehen. Weiter wird dem Wasser-Objekt ein 2D Wasser-Shader hinzugefügt. Er erzeugt eine windbewegte animierte Wasseroberfläche.

2.8.2 Rauch / Feuer

Für die Entwicklung von Feuer und anderen atmosphärische Effekte stellt Cinema 4D das Plug-In „PyroCluster“ zur Verfügung. „PyroCluster-Effekte sind echte 3D-Effekte, die auf Cinema 4Ds Partikelsystem basieren [7].“ Das Arbeiten mit PyroCluster ist trotz eigenem Manuel und auf dem Internet gefundener Anleitung nicht dem Erfolg beschert gewesen. Im Material-Editor PyroCluster sind derart viele Einstellungen möglich, dass bei der kleinsten Änderung die grössten Veränderungen auftreten. Die Vielzahl der Einstellungen führte nicht zuletzt aus zeitbegrenzter Überlegungen auch dazu, dass nicht alle Optionen untersucht und begriffen werden konnten.

2.8.3 Menschen

Unter dem Menü Objekt Objekt-Bibliothek gibt es drei Grundtypen Mensch (Child, Men, Women). Die vorgegebene Baustruktur des Objektes lässt es zu, dass nur einzelne Objektteile wie zum Beispiel der Arm nicht in unmögliche Positionen gedreht werden kann. Wenn der Arm an der Schulter gedreht wird, kommt der ganze Unterarm mit.
Der Bewegungsablauf von Menschen ist die Königsklasse der Animation und wurde daher nicht näher untersucht.

3 Datenfluss

Verfasser: D. Emmenegger

Bisher wurden an der FHBB einige Visualisierungsprojekte durchgeführt. Nur wurde bis jetzt nie ein klarer Datenfluss definiert. So musste eigentlich bei jedem Projekt wieder neu überlegt werden, wie die genaue Arbeitsabfolge ist. Das führte natürlich zu unerwünschten Zeiteinbussen und Mehrspurigkeiten.

Mit den Erfahrungen aus den bereits erstellten Projekten soll nun ein klarer Datenfluss erstellt werden, damit bei zukünftigen Projekten dieser angewendet werden kann. Somit sollten keine Mehrspurigkeiten und unnötige Zeitverluste mehr auftreten.

3.1 Momentane Situation

Es wurde aus den bisherigen 3D-Projekten der Klassen G2001 und G2002 ein Datenfluss-diagramm erzeugt, welches die bisherigen Abläufe wiedergibt. Für die Herstellung dieses Diagramms wurden insbesondere folgende Projekte verwendet:

- Schloss Wildenstein
- Ruine Homburg
- Alpsiedlung Vanescha

Da jedes Projekt seine speziellen Eigenschaften besitzt, wurde sehr schnell klar, dass bei jedem der Datenfluss verschieden war. Da sich die Unterschiede aber in einem vertretbar kleinen Rahmen halten, konnte für alle Projekte ein gemeinsamer Datenfluss erarbeitet werden.

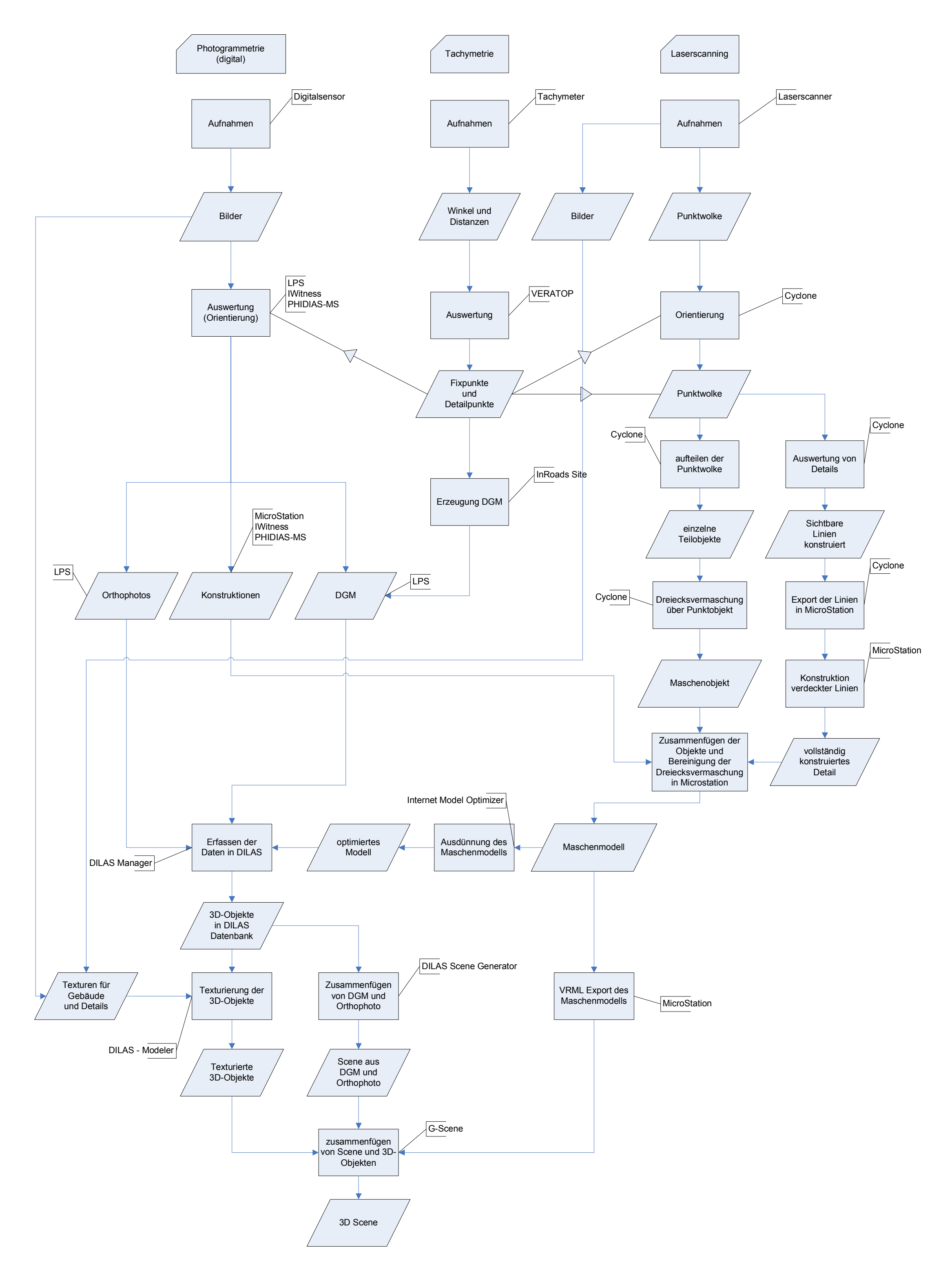


Abbildung 7 Bisheriger Datenfluss

3.2 Analyse

Betrachtet man diesen Ablauf, so fällt sofort auf, dass er komplex ist. Auch wird erkennbar, dass man mit nur einer Aufnahmemethode nicht zum Ziel kommt. Es müssen also immer mehrere Aufnahmemethoden kombiniert werden, was die Arbeit nicht gerade erleichtert. Aus dieser Kombination der Aufnahmemethoden folgt unweigerlich eine Fülle von Auswerteprogrammen und Datentransfers. Diese Fülle von Auswerteprogrammen kann nicht vermindert werden, da meist verschiedene Hersteller hinter den Programmen stehen und diese kein Interesse an einem einheitlichen Auswerteprogramm haben. So wird sich der Auswerteprozess nicht vereinfachen lassen.

Trotzdem kann man den Ablauf in drei Teile aufsplittern:

- Modellierungsteil
- Texturierungsteil
- Animierungsteil

Da wir mit Cinema 4D zusätzlich noch Videos von der gesamten Szene erstellen, wird dieser Punkt noch zusätzlich hinzugefügt. Somit lässt sich der gesamte Ablauf eines Projektes in vier Teile zerlegen. Auch werden damit die Schnittstellen zwischen den einzelnen Teilen aufgezeigt. Bis jetzt waren diese meistens nicht definiert. Wie genau der Austausch der Daten zu erfolgen hatte, war nicht klar geregelt.
Der ganze Ablauf wird in Phasen unterteilt, wie man es auch in modernen Projektmanagement Systemen kennt. Dies soll ein phasenweises Arbeiten ermöglichen.

3.3 Lösungskonzept

Hier sollen die einzelnen Phasen der Projekte und die Schnittstellen zwischen den Phasen beschrieben werden. Als Grundlage für die Lösungen dienen die beiden Projekte Wildenstein und Homburg. Es wurden als erstes die einzelnen Teilschritte eines Projektes in die Phasen eingeteilt. Somit entstand ein grober Ablauf eines möglichen Projektes.

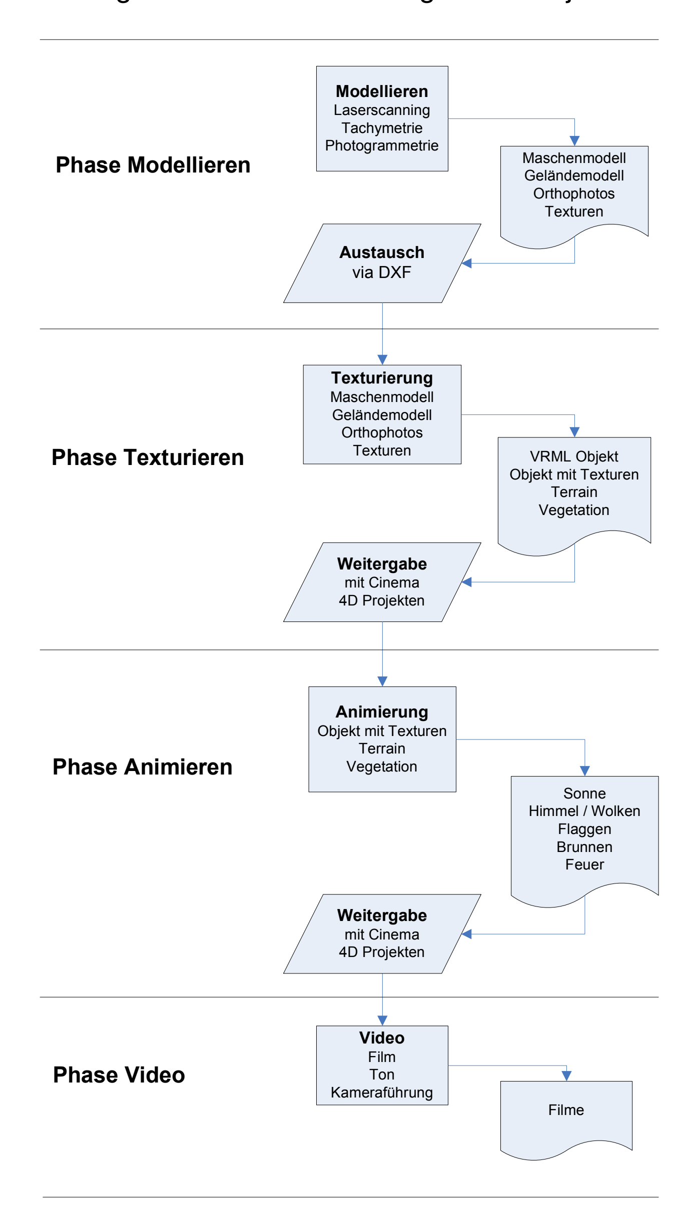


Abbildung 8 Grober Ablauf eines Projektes

Wichtig dabei ist, dass der Datenaustausch jeweils zur vorhergehenden Phase gehört. Somit sind die Personen, welche die Phase ausgeführt haben, auch für den Transfer verantwortlich. Gerade bei einem Projekt wie Wildenstein oder Homburg, bei welchem relativ viele Personen mitwirkten, ist dies sehr wichtig. Denn nur die Personen, welche die Daten erstellt haben, können diese beim Import und Export auf ihre Richtigkeit kontrollieren. Weiter ist es unbedingt nötig, einen Phasenverantwortlichen festzulegen. Es geht nicht in erster Linie darum, jemandem die Verantwortung zuzuschieben. Viel wichtiger ist, dass man bei Problemen genau **eine** Ansprechsperson hat und nicht eine ganze Gruppe. Dies kann die Lösung von Problemen um ein Vielfaches beschleunigen.

Anschliessend wurden die einzelnen Phasen genauer untersucht und beschrieben. Infolge dessen wurden häufig die beiden Begriffe "Grundlagendaten" und "Schnittstelle" verwendet. Um Klarheit über die Bedeutung dieser Begriffe zu schaffen, hier eine Erklärung:

Austauschdaten

Darunter sind die eigentlichen Daten gemeint, welche ausgetauscht werden. Sie haben nichts mit dem Format, welches zum Austausch verwendet wird, zu tun.

Schnittstelle

Damit ist die Art des Austausches der Daten gemeint. z.B., ob die Daten via DXF-File oder als ASCII-File ausgetauscht werden.

3.3.1 Phase Modellierung

Diese Phase ist mit Abstand die grösste und aufwändigste. Sie enthält die Erhebung der Daten, wie auch die Weiterverarbeitung zu den Austauschprodukten.

Wir befassen uns hier mit der Definition der Austauschprodukte und den Schnittstellen da bei den Projekten Wildenstein und Homburg die eigentliche Erhebung der Daten abgeschlossen ist. Für Details, wie man zu diesen Produkten kommt siehe Kapitel 8.

3.3.1.1 Austauschdaten

Um die Phase des Texturierens ermöglichen zu können, braucht es Grundlagendaten, welche man in dieser Phase erstellen muss. Diese Daten umfassen im Detail:

- Maschenmodell
- Geländemodell
- Orthophotos
- Texturen

Maschenmodell

Das Maschenmodell ist das Modell des Objektes, welches man visualisieren will. Es wurde aus verschiedenen Aufnahmemethoden erstellt, und stellt das eigentliche Hauptergebnis der Phase Modellierung dar. Ausgetauscht wird es am besten mit dem DXF-Format, da die Konstruktion meist in einem CAD stattfindet. Bei der Konstruktion und dem Datentransfer sind jedoch einige Punkte zu beachten:

- Es ist eine logische Layerstruktur zu erstellen.
- Die einzelnen Teile eines Objektes sind schon beim Konstruieren zu unterteilen. z.B. ist eine Hausfassade in ihre Fassadenteile aufzusplittern. Dies erleichtert anschliessend die Texturierung.
- Das Maschenmodell ist so weit wie möglich auszudünnen.
- Es ist ein Koordinatenshift anzubringen, so, dass die Koordinatenwerte klein sind. Diesen benötigt man, um in Cinema 4D angenehm arbeiten zu können. (Kap. 2.2)
- Es dürfen keine komplexen Objekte zur Konstruktion verwendet werden. Diese werden sonst beim Import in Cinema 4D ignoriert und müssen erneut konstruiert werden.

Geländemodell

Es ist möglich, dass ein Geländemodell von der Umgebung bereits existiert und übernommen werden kann. Allerdings sollte man in diesem Fall besondere Aufmerksamkeit der Anpassung des Geländemodells an das Objekt schenken, da eine nachträgliche Korrektur aufwändig ist.
Auch hier muss der gleiche Koordinatenshift angebracht werden wie beim Maschenmodell.
Dieser ermöglicht ein einfaches Zusammenfügen des Orthophotos und des Maschenmodells in Cinema 4D.

Orthophotos

Sie werden oft in Verbindung mit einem Geländemodell verwendet und dienen als Landschaft zum eigentlichen Objekt. Deshalb sollte die Grösse des Orthophotos gleich sein wie die des Geländemodells.
Ein Koordinatenshift ist hier nicht unbedingt notwendig. Es besteht die Möglichkeit das Orthophoto auf das Geländemodell zu mappen. (Kap. 4.2.2)

Texturen

Will man ein Objekt fotorealistisch darstellen, dürfen Fotos zum Texturieren nicht fehlen. Meist entstehen die Fotos in Verbindung mit einem Nahbereichsphotogrammetrieprojekt. Daher existieren für solche Fotos Orientierungswerte, welche in Cinema 4D verwendet werden können. Die Fotos müssen relativ viele Anforderungen erfüllen:

- Die Nummerierung der Bilder muss eindeutig sein.
- Nur Bilder mit Orientierungswerten können einfach gemappt werden. (Kap. 4.2.1)
- Die Mastertabelle ist vollständig auszufüllen. (Kap. 4.2.1.3)
- Bei den Bildern muss die Verzeichnung korrigiert sein. (Kap. 4.2.1.4)
- Die Texturen sollen als JPEG-Files abgespeichert sein.
- Ein allgemeiner radiometrischer Abgleich muss stattgefunden haben. (Kap. 4.2.5)

Besonders ist bei der Aufnahme von Bildern auf einige Punkte zu achten da sie nicht nur zum Auswerten, sondern auch zum Texturieren verwendet werden.

- Fassaden müssen mindestens in einem Foto vollständig sichtbar sein.
- Es ist eine möglichst rechtwinklige Aufnahme zum Objekt anzustreben.
- Schattenwürfe und andere störende Einflüsse sind zu minimieren.
- Aufnahmen sind nach dem Grundsatz "Lieber ein Foto zuviel, als eines zuwenig" auszuführen.

3.3.1.2 Schnittstellen

Um diese Phase abzuschliessen, ist ein Datenaustausch zwischen den Phasen nötig. Da die Daten in der Modellierungsphase meist einzeln angefertigt werden, ist dies in Cinema 4D beizubehalten. Es soll für jede der Grundlagendaten ein eigenes Cinema 4D Projekt erstellt werden. Anschliessend kann man sich seine Szene nach dem Baukasten-Prinzip selber zusammenstellen. Hier sollen die nötigen Vorgänge beschrieben werden.

Maschenmodell

Sofern das Modell keine komplexen Objekte enthält, ist ein Transfer via DXF-File zu Cinema 4D unproblematisch. Leider gibt es aber keine 100%ige Sicherheit. Deshalb muss das importierte Modell in Cinema 4D visuell überprüft werden.

Geländemodell

Das einfachste Verfahren, um ein Geländemodell einzulesen ist, wenn man dieses zuerst ins CAD (Microstation) importiert, und anschliessend dies als DXF-File in Cinema 4D exportiert. Dieser Umweg ist leider nötig, da es sich bei Cinema 4D um kein "Geomatikprodukt" handelt und Geländemodelle nicht importiert werden können.

Orthophotos

Wie oben schon gesagt, ist eine Georeferenzierung des Orthophotos nicht notwendig. Daher genügt es, das Orthophoto als einfaches TIF-File zu speichern.

Texturen

Da die Texturfotos von der Auswertung her schon verzeichnungskorrigiert und als JPEG-File abgespeichert sind, beschränkt sich dieser Austausch auf das Kopieren der Bilder.

3.3.2 Phase Texturierung

Nachdem nun der Datenimport in Cinema 4D erfolgreich beendet ist, kann mit dem Start der nächsten Phase begonnen werden. Bisher war das Texturieren von grossen komplexen Objekten ein nicht gelöstes Problem, da jedes einzelne Polygon separat texturiert werden musste. Mit Cinema 4D steht uns aber das Kameramapping zur Verfügung, welches dieses Problem löst. (Kap. 4.2.1)

3.3.2.1 Austauschdaten

Hier findet zwischen den Phasen kein eigentlicher Datenaustausch statt, da die Texturierung und die Animierung im selben Programm stattfinden. Es ist trotzdem wichtig, dass die Produkte klar definiert werden. Schlussendlich sollen drei Cinema 4D Projekte übergeben werden:

- Objekt (Maschenmodell und Texturen)
- Terrain (Geländemodell und Orthophotos)
- Vegetation

Maschenmodell und Texturen

Das Modell soll soweit als möglich mit Texturen versehen sein. Es soll ein einzelnes Cinema 4D Projekt mit dem texturierten Modell gespeichert werden. Je nach Datengrösse der Texturen sind diese noch zu resamplen.

Beim Texturieren gibt es ein paar allgemeine Regeln zu beachten:

- Zuerst das ganze Objekt texturieren und erst anschliessend die radiometrischen Feinkorrekturen vornehmen.
- Nur Bilder, welche auch in der Auswertung gebraucht werden, können für ein "einfaches" Texturieren verwendet werden.
- Möglichst viel aus dem gleichen Foto texturieren, so werden unschöne Schnittstellen minimiert.
- Nur Objekte, welche ca. den gleichen Abstand zum Brennpunkt haben, aus dem gleichen Foto texturieren.

Um einen VRML Export zu ermöglichen, müssen die Texturen auf dem Objekt fixiert werden. Dies wird in Cinema 4D mit einem so genannten "Tag" verwirklicht. (Kap. 2.1.2)

Geländemodell und Orthophoto

Wie die beiden Objekte in Cinema 4D zusammengefügt werden können, wird in Kap. 4.2.2 erläutert.

Vegetation

Besonders bei der Vegetation ist es wichtig, ein eigenes Projekt zu erzeugen. Die Datenmenge übersteigt sehr schnell die Gesamtgrösse der anderen Projekte. Wie genau die Vegetation modelliert wird, ist in Kapitel 5 beschrieben.

3.3.2.2 Schnittstellen

Da es sich in dieser Phase nicht um einen eigentlichen Datentransfer handelt, werden keine Schnittstellen benötigt. Es werden einfach die erstellten Cinema 4D Projekte ausgetauscht.

3.3.3 Phase Animierung

Die bis jetzt erstellte Szene in Cinema 4D ist nur statisch. Es wäre möglich, einen Film daraus zu erstellen, nur würde sich darin ausser der Kamera nichts bewegen. Um nun die Realität zu steigern, werden zusätzlich noch Objekte animiert. Typische Objekte zum Animieren sind:

- Himmel / Wolken
- Sonne
- Flaggen
- Brunnen
- Türen
- Menschen

Ein besonderes Merkmal verdient hierbei die Gruppe "Menschen". Da es extrem schwierig ist, Bewegungsabläufe realitätsnah wiederzugeben, wird meist eine statische Darstellung bevorzugt.
Wie solche Animierungen erzeugt werden, wird in Kap. 2.8 beschrieben.

3.3.3.1 Austauschdaten

Auch hier gibt es wieder nur die Cinema 4D Projekte als Austauschdaten. Grundsätzlich sollten die animierten Effekte in das Objekt integriert werden. Die Integration bleibt aber abhängig von der Grösse der Animierungseffekte. Bei sehr grossen Effekten kann es durchaus besser sein, wenn man ein separates Cinema 4D Projekt eröffnet.

3.3.3.2 Schnittstellen

Da es sich in dieser Phase nicht um einen eigentlichen Datentransfer handelt, werden keine Schnittstellen benötigt. Es werden einfach die erstellten Cinema 4D Projekte ausgetauscht.

3.3.4 Phase Video

Aus der bis jetzt erstellten Szene in Cinema 4D können ohne Probleme einzelne Bilder erzeugt werden. Hängt man nun sehr viele dieser Bilder aneinander, so entsteht ein Film. Genau nach diesem Prinzip werden Filme mit Cinema 4D hergestellt. Zwar können die Bilder nicht mit diesem Programm zusammengesetzt werden, aber es empfiehlt sich, Einzelbilder zu rendern. Gründe, welche für dieses Vorgehen sprechen sind:

- Bei einem Systemabsturz sind die erstellten Bilder nicht verloren.
- Nachträgliche kleine Änderungen können vorgenommen werden (Alpha Kanal).
- Die Qualität der Bilder ist höher als beim Film.

Einige Punkte, welche zusätzlich noch beachtet werden müssen, sind im Kapitel 6 nachzusehen.

3.3.4.1 Austausch zum Videoprogramm

Es werden nur einzelne Bilder dem Videobearbeitungsprogramm übergeben. Diese müssen allerdings der Reihe nach nummeriert sein, damit man sie richtig importieren kann.

3.3.4.2 Schnittstellen

Schnittstellen sind hier nicht vorhanden, da es keine weiteren Phasen mehr gibt.

3.4 Schlussfolgerungen

Es wurde ein Erstellungsablauf zum Modellieren von 3D-Geoobjekten geschrieben. Der Ablauf wurde dabei in einzelne Phasen aufgeteilt, wie es heutzutage beim Projektmanagement üblich ist. Da dies ein Erstlingswerk ist, wird dieser Ablauf sicherlich noch ergänzt oder geändert werden. Aber eine solide Grundlage konnte erstellt werden. Die letztendliche Tauglichkeit dieses Ablaufes muss in der Praxis jedoch noch bewiesen werden.
Mit der Verwendung dieses Ablaufes kann die Zeit, welche zur Erstellung von komplexen 3D-Geoobjekten benötigt wird, stark verkürzt werden. Man muss nicht mehr bei jedem Projektstart zuerst einen eigenen Projektablauf definieren, wie es bis jetzt gemacht wurde.

Der ganze Ablauf ist noch auf die verwendeten Programme zugeschnitten. Eine programm-unabhängige Formulierung wäre noch zu überlegen. Dies würde aber einen viel höheren Abstraktionsgrad voraussetzen, bei welchem ein praktischer Nutzen für die Praxis fragwürdig wäre.

Die Phase Video fiel hier relativ knapp aus. Eine Vertiefung dieser Phase wäre empfehlenswert. Da dies aber nicht unser Fachgebiet ist, ist eine Zusammenarbeit mit der Projektgruppe "Architektur und Film" der HGK Basel anzustreben. Eine erste Zusammenarbeit fand im Rahmen dieser Diplomarbeit bereits statt. In Zusammenarbeit wurde der Film des Schlosses Wildenstein erzeugt. Diese Arbeit zeigte eindrucksvoll, wie wichtig eine interdisziplinäre Zusammenarbeit sein kann, da wir uns mit der Produktion von Videos in für uns völlig neue Gebiete wagten.
Wir dürfen auch nicht vergessen, dass wir die Thematik nicht neu erfinden.

4 Texturen

Verfasser: D. Emmenegger [2]

4.1 Theoretische Grundlagen

Schaut man sich im Raum ein bisschen um, so bemerkt man sehr schnell ein grosses Spektrum an geometrischen Körpern.
Um nun diese aber realitätsgetreu abbilden zu können, reicht es nicht, einfach diese geometrischen Formen nachzubilden. Vielmehr ist der visuelle Eindruck, wie z.B. Farbe, oft wichtiger als die eigentliche Form. Es ist nicht nur die Farbe, welche für eine realitätsnahe Abbildung benötigt wird. Auch Lichteffekte, Betrachtungswinkel und Projektionsarten haben einen grossen Einfluss.

Texturen bieten nun die Möglichkeit, das Erscheinungsbild von Objekten komplexer zu gestalten. Die Oberfläche eines Würfels kann durch eine Ebene modelliert werden. Das Texturieren dieser Ebene kann als Aufbringen eines Bildes auf die Würfeloberfläche angesehen werden. Dies wird im allgemeinen als Texturierung verstanden.

Die Zuweisung der 2D-Texturkoordinaten zu den 3D-Objektkoordinaten kann somit folgend beschrieben werden:

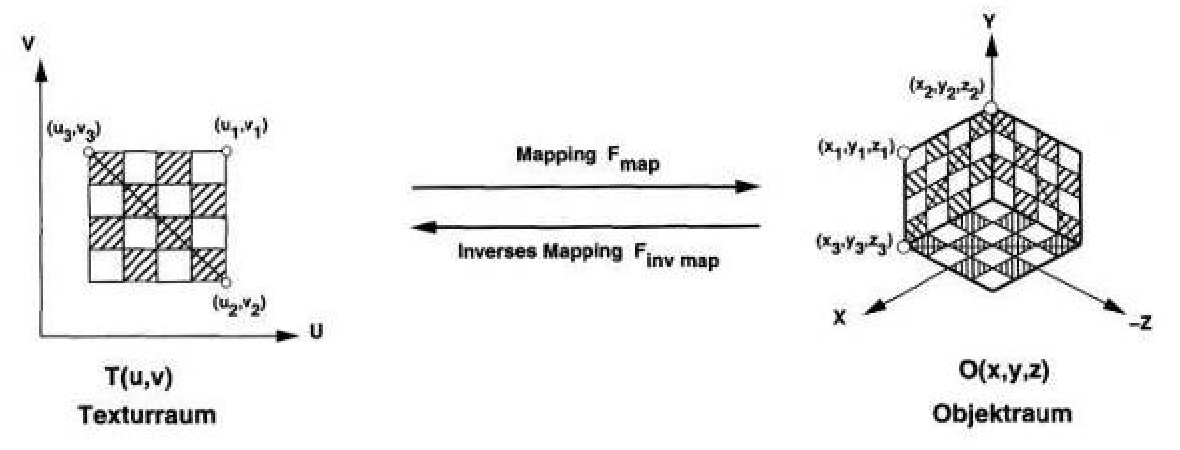


Abbildung 9 Mapping und invers Mapping [2]

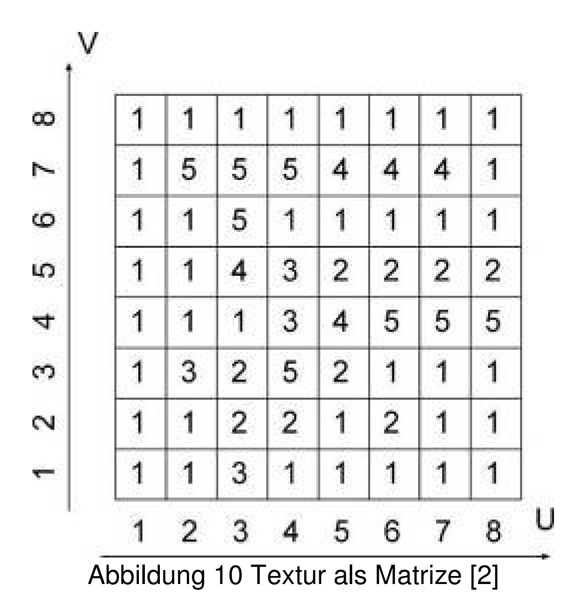
Bei der Visualisierung wird die so genannte "inverse Mapping" Funktion verwendet. Würde man die "normale Mapping" Funktion gebrauchen, würde es zu Mehrfachzuweisungen und Löschungen von einzelnen Pixel kommen. Dieses Problem ist aus der digitalen Bildverarbeitung bekannt.

4.1.1 Arten von Texturen

Grundsätzlich sind zwei Arten von Texturen zu unterscheiden:

- diskrete Texturen
- prozedurale Texturen

Eine diskrete Textur ist grundsätzlich ein Array, in unserem zweidimensionalen Beispiel also eine Matrize. siehe Abbildung 10



V \ U	1	2	3	4	5	6	7	8
8	1	1	1	1	1	1	1	1
7	1	5	5	5	4	4	4	1
6	1	1	5	1	1	1	1	1
5	1	1	4	3	2	2	2	2
4	1	1	1	3	4	5	5	5
3	1	3	2	5	2	1	1	1
2	1	1	2	2	1	2	1	1
1	1	1	3	1	1	1	1	1

Abbildung 10 Textur als Matrize [2]

Die Einträge können je nach Verwendung nun verschiedene Bedeutungen haben. Es können z.B. Graustufenwerte oder Höheninformationen sein. Meistens werden aber Informationen über die Farbe des jeweiligen Texels (TEXture-ELement) gespeichert. Die Arrays weisen dabei meist eine Einheitsgrösse von 1x1 auf. Somit ist eine unabhängige Adressierung der Texel möglich, was eine Nachbearbeitung der Texturen vereinfacht. Ebenso können die Auflösungen der Texturen verändert werden, ohne dass die Adressierung angepasst werden muss.



Abbildung 11 Vergleich mit Textur und ohne Textur

Am Beispiel des Tennisballes sieht man sehr schön, was eine Textur zur Realitätsnähe beiträgt. Der linke Ball könnte ebenso gut einen Basketball darstellen; erst die Textur macht die Identifizierung eindeutig.

Wenn man sich die Textur aber ein wenig näher betrachtet, fällt auf, dass es sich nicht nur um ein aufgeklebtes Foto handelt. Es wurde zusätzlich ein Tiefeneindruck geschaffen, welcher zusätzlich zur Realitätsnähe beiträgt. (siehe auch Kap. 4.1.5.3 und Kap. 4.1.5.4)

Im Gegensatz zu den diskreten Texturen werden bei den prozeduralen Texturen die einzelnen Farbwerte nicht gespeichert, sondern anhand einer Funktion jedes Mal neu berechnet.

Vor und Nachteile der Texturarten:

Texturart	Vorteil	Nachteil
diskrete Texturen	• Einfache Erfassung mit Fotoapparat oder Scanner möglich. • Komplexe 2D Texturen sind einfach zu erstellen.	• Texturwerte im Objektraum müssen interpoliert werden. • Für Texturwerte ausserhalb des Wertebereichs müssen Sonder-regeln getroffen werden. • Je nach Bildqualität grosser Speicherbedarf. • Probleme bei der radio-metrischen Gleichheit der Bilder (Sonne, Schattenwurf). • Beim Mapping auf beliebige Flächen treten Verzerrungen auf (Aliasing-Problem).
prozedurale Texturen	• Speicheraufwand minimal • Texturwerte an jeder Stelle mit optimaler Genauigkeit • Texturen für gesamten Raum definiert	• Komplexe Texturen müssen mathematisch beschrieben werden.

Tabelle 1 Vergleich der Texturarten

4.1.2 Beleuchtungsmodelle

Unter Beleuchtungsmodellen versteht man in der 3D-Visualisierung Arten, wie das Licht simuliert wird. Es werden grundsätzlich zwei Arten von Modellen unterschieden: Lokal- und Globalmodelle.
Meist wird aber unter dem Begriff "Beleuchtungsmodell" ein lokales Modell verstanden.

4.1.2.1 Lokale Modelle

Bei lokalen Beleuchtungsmodellen werden die Vorschriften zur Berechnung der Farb- und Grauwerte der Einzelnen Bildpunkte gemacht. Die Grundlagen dieser Berechnungen sind die Gesetzte der Physik. Das Modell muss also die Ausbreitung des Lichts in einer definierten Umgebung, mit einem vernünftigem Rechenaufwand bereitstellen können.

Die gängigsten Methoden sind:

- Konstante Beleuchtung
- Gouraud-Shading
- Phong-Shading

4.1.2.2 Globale Modelle

Im Gegensatz zu lokalen Modellen wird in globalen Modellen die Lichtausbreitung in einer ganzen Szene berechnet. Es wird auch nicht versucht, mit einem möglichst schnellen Algorithmus ein möglichst gutes Resultat zu erzielen. Vielmehr steht die realitätsnahe Wiedergabe von Lichtverhältnissen im Vordergrund. Dies hat zur Folge, dass eine Darstellung mit einem globalen Modell sehr viel rechenintensiver ist als eine Darstellung mit einem lokalen Model.

Die gängigsten Methoden dazu sind:

- Raytracing (Strahlenverfolgung)
- Radiocity

4.1.3 Ablauf Texturierung (Textur – Mapping)

Würde man jetzt einfach eine 2D-Textur auf ein 3D-Objekt aufziehen, wäre das Ergebnis etwa folgendermassen:

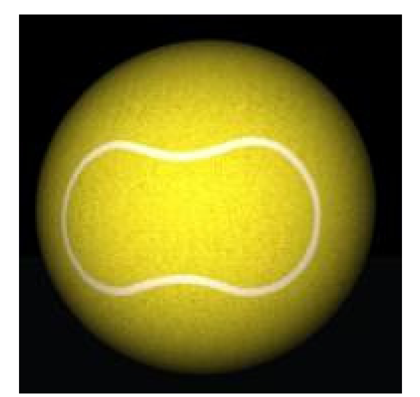


Abbildung 12 Textur ohne Projektion

Das Bild entspricht ganz dem Original, was aber nicht ganz erwünscht ist. Man möchte eigentlich, dass die Textur die Form eines Balles nachahmt

Dies passiert deshalb, weil die Textur noch nicht perspektivisch korrekt auf die Kugel abgebildet wird. Man braucht also eine Funktion, welche zu gegebenen 3D-Körperkoordinaten die 3D-Texturkoordinaten berechnet.

Zu texturierende Flächen werden meist als Polygone angenähert. Diese Polygone lassen sich weiter auch ohne Problem in Dreiecke zerlegen. Eine solche Projektionsfunktion für ein einzelnes Dreieck könnte somit folgendermassen aussehen:

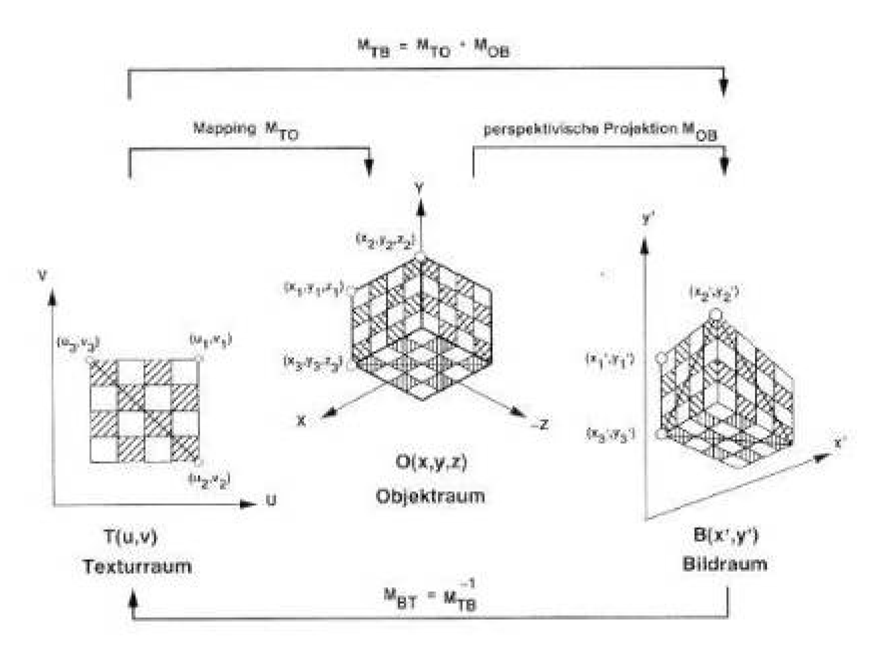


Abbildung 13 Transformation zwischen Textur,- Objekt- und Bildschirmkoordinaten [2]

Zuerst werden die Texturkoordinaten (u, v) mit einer Mappingfunktion in den Objektraum abgebildet. Jetzt befindet sich die Textur zwar auf dem Objekt, nur stimmt die Perspektive noch nicht. Hierzu werden die Objektkoordinaten mit einer perspektivischen Projektion angepasst. Somit erhalten wir die perspektivisch richtigen Bildraumkoordinaten.

Zum erklären der Funktion wurde der Weg des normalen Mappings verwendet. In der Praxis wird aber wie früher schon erwähnt, die inverse Mappingfunktion gebraucht. Dies führt zu keinen grossen Schwierigkeiten, da die Funktion aus Matrizen bestehen, welche sich einfach invertieren lassen.

4.1.4 Probleme bei der Texturierung

Jetzt wo der Ablauf einer Texturierung bekannt ist, zeigen sich auch schon die ersten Probleme. Wie gesagt, wird für das Texturieren die inverse Funktion gebraucht. Nun ist diese Funktion aber nur für die Werte definiert, welche in der Original 2D-Textur vorhanden ist. Da die inverse Funktion in den seltensten Fällen gerade Texturkoordinaten ergeben, stellt sich die Frage: Woher nun diese Werte nehmen? Man könnte einfach die Auflösung der Texturen erhöhen, was aber zu einem rasanten Anstieg des benötigten Speichers führen würde. Einfacher ist, man interpoliert sich die gewünschten Informationen. Die verschiedenen zur Verfügung stehenden Interpolationsverfahren sind:

- nächste Nachbarschaft (nearest neighbour)
- billineare Interpolation

Wobei bei der "nächste Nachbarschaft"-Interpolation einfach der nächst liegende Pixelwert genommen wird. Dies kann zu störender Blockbildung führen. Diese Methode liefert nicht die gewünschte Qualität. Besser geeignet ist für diesen Zweck die "billineare Interpolation". Sie ist zwar rechenintensiver, liefert aber wesentlich bessere Resultate. Dabei werden die dem gesuchten Punkt nächsten vier Farbwerte, gemäss ihrem Abstand gewichtet und zu einem neuen Farbwert addiert.

Leider löst dieses Verfahren nicht alle Probleme, welche bei perspektivischen Projektionen entstehen.
Es gibt weitere Probleme:

- Welcher Farbwert soll dargestellt werden, wenn in einiger Entfernung zum Augpunkt mehrere Texelwerte auf einen Pixelwert fallen?
- Auch ist die detaillierte Darstellung von weit entfernten Objekten nicht sinnvoll. Da die Objekte meistens nicht ganz sichtbar sind, wäre eine entfernungsabhängige Reduktion der Pixelwerte erstrebenswert. (Kap. 4.1.6.1)

Ausserdem gibt es Probleme, wenn das texturierte Objekt animiert wird. Dann findet nämlich die Interpolation zu neuen Punkten ständig statt. Eine Fläche die eigentlich nur verdreht wurde, bekommt so plötzlich eine andere Farbe.

Ein Problem welches nur bei den diskreten Texturen auftritt. Was passiert, wenn die Textur zu Ende ist, und der zu füllende Platz aber noch weitergehen würde?
Dazu gibt es verschiedene Lösungsansätze:

- Es kann ein konstanter Farbwert für die restliche Fläche vergeben werden. Somit ist die Grösse der Textur auch klar ersichtlich.

- Man kann auch den ersten und den letzten gültigen Farbwert festhalten. Man spricht dann von so genanntem "Texture clamping".

- Eine dritte und wahrscheinlich am meisten verbreitete Methode ist die periodische Wiederholung der Textur. Die Textur wird dann einfach am Ende wieder angefügt. Somit entsteht eine flächenfüllende Textur.

4.1.5 Projektionsarten (Mapping-Arten)

Der Ablauf der Texturierung sollte nun bekannt sein. Wie und mit welcher Projektion nun die vorhandenen Texturen dargestellt werden, wird in diesem Kapitel besprochen.

4.1.5.1 parametrisierte Flächen

Darunter werden parametrisierte Flächen wie z.B. Bézier oder B-Spline-Flächen verstanden.
Der einfachste Fall bei Texturierung von parametrisierten Flächen ist die direkte Zuordnung der Flächenpunkte zu der Texturkoordinate. Falls nötig, kann die Zuordnung auch über eine affine oder andere Abbildung erfolgen.

Leider ist die direkte Texturierung von solchen Flächen sehr rechenintensiv. Darum werden diese Flächen meistens durch Dreiecke ersetzt, was einen Zeitgewinn erbringt. Somit kann wieder das Verfahren wie unter Kap. 4.1.3 angewendet werden.

4.1.5.2 komplexe Objektflächen (Zylinder, Kugel, Box)

Der Texturierungsablauf, welcher im Kapitel 4.1.3 erklärt wurde, setzt voraus, dass die Zuordnung der Punkte im Texturraum zum Objektraum bekannt sei. Bei einem Modell mit wenigen Dreiecken kann der Benutzer diese Zuordnung noch manuell vornehmen. Je komplexer aber das Objekt ist, desto schwieriger und zeitintensiver wird diese Zuordnung.
Es gibt ein Verfahren zur Lösung dieses Problems. Die Grundidee besteht in einer Zweiteilung der Projektion.

1. Der Körper wird mit einer passenden parametrisierbaren Fläche umhüllt. Die 2D-Textur wird in einem ersten Schritt auf diese Fläche abgebildet.

2. In einem zweiten Schritt erfolgt die Projektion vom Hilfskörper auf das zu texturierende Objekt.

Die gängigsten Hilfskörper sind dabei Zylinder-, Kugel-, Flächen-, und Quaderoberflächen.

4.1.5.3 Bump Mapping

Mit dieser Technik ist es möglich einen Körper auf einfache weise runzlig, rauh, zerknittert oder gekräuselt darzustellen.
Im Wesentlichen besteht die Idee, nicht den Körper geometrisch zu verformen; vielmehr will man durch Schattierung der Oberfläche einen räumlichen Eindruck vermitteln. Deswegen ist es Voraussetzung, bei dieser Darstellungsart ein Beleuchtungsmodell zu verwenden.
Am besten zeigt man die Wirkung an einem Beispiel:

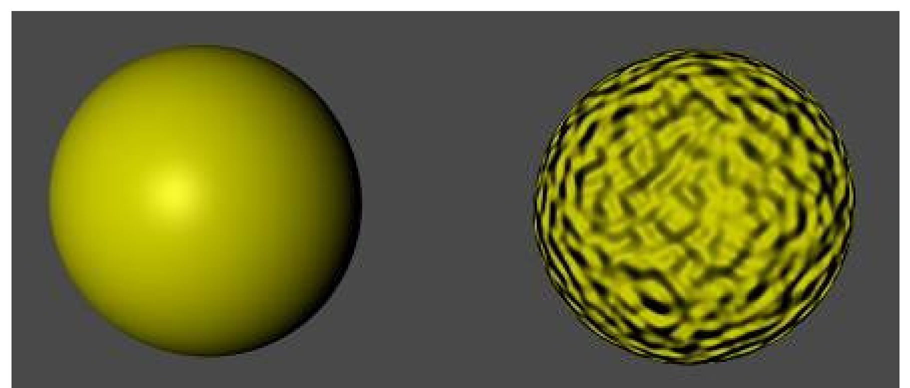


Abbildung 14 Links ohne, rechts mit Bump Mapping

Meist werden zur Definition der Schattierung Höheninformationen verwendet, welche als Graustufenbild gespeichert sind (Heightmapping). Normalerweise repräsentiert dabei Schwarz (Farbwert 0) die "tiefste“ Stelle und Weiss (Farbwert 255) die "höchste" Stelle. Eine weitere Methode zur Definition der Schattierung ist das so genannte Normalmapping. Hier werden an Stelle von Höheninformationen Vektoren im RGBFarbentripel konvertiert. Der Vektor muss nur noch normalisiert werden, um die Helligkeitsinformationen für den betreffenden Punkt zu bekommen.

4.1.5.4 Displacement Mapping

Die Idee hierbei ist die gleiche wie beim Bump Mapping: Man will die Oberfläche eines Körpers runzlig, rauh, zerknittert oder gekräuselt darstellen. Der Unterschied besteht aber in der Art der Verformung. Beim Bump Mapping wird nur der Schein einer Verform dargestellt, der Körper aber behält seine eigentliche Form. Ganz anders beim Displacement Mapping: Hier wird der Körper wirklich verformt. Diese Verformung basiert auch auf einem Relief, welches aber meist aus der eigentlichen Textur gebildet wird.

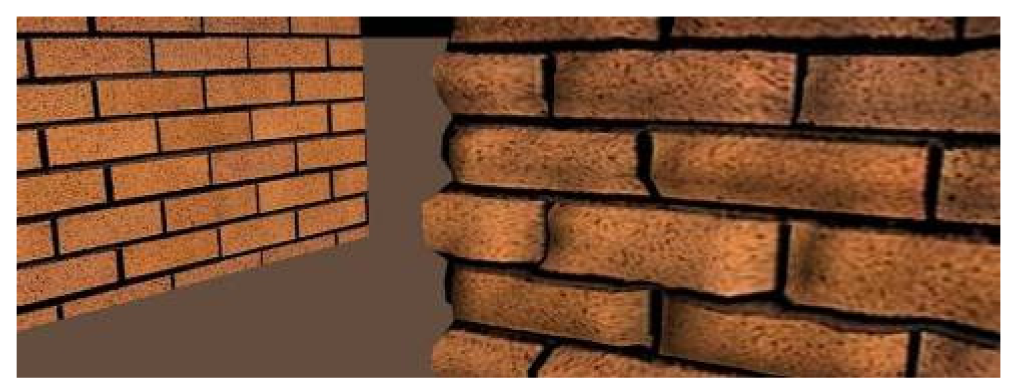


Abbildung 15 Hintergrund mit Bump Mapping, Vordergrund mit Displacement Mapping

Sehr schön sieht man den Unterschied zwischen Bump- und Displacementmapping in diesem Beispiel. Im Vordergrund wird die Wand wirklich verformt, während sich der im Hintergrund befindliche Block nicht verformt.

4.1.5.5 Reflection Mapping

Bis jetzt wurde nie der Einfluss von Reflexionen auf die Textur berücksichtigt. Eine Möglichkeit dazu bietet das Reflection Mapping.
Die Grundidee des Reflection Mapping zeigt die Abbildung 16. Eine gesamte Szene wird hier durch ein einzelnes Objekt repräsentiert. (Zu beachten gilt es, dass dies wieder eine Vorwärtsabbildung ist. Beim eigentlichen Mapping geschieht die Berechnung aber invers.) Dieses Objekt wird durch eine virtuelle Kugel eingeschlossen. Auf der Innenseite der Kugel wird die 2D-Szenenumgebung (Reflexionstextur) aufgetragen. Diese Textur nennt man Reflection Map. Diese Map wird als Intensitätsfeld gespeichert und über die Kugelkoordinaten adressiert.

Bei der Berechnung der Beleuchtung kann nun einfach in einem Objektpunkt **P** die ideal reflektierte Blickrichtung **R** berechnet werden. Ist jedoch der Kugelradius viel grösser als die Strecke **P**-**W**, kann auf eine Schnittberechnung **R**, **Kugel** verzichtet werden. Man kann direkt die Kugelkoordinaten von **R** bestimmen und damit direkt die Reflection Map indizieren. Man verwendet also den Schnittpunkt **V**, **Kugel** anstatt **R**, **Kugel**.

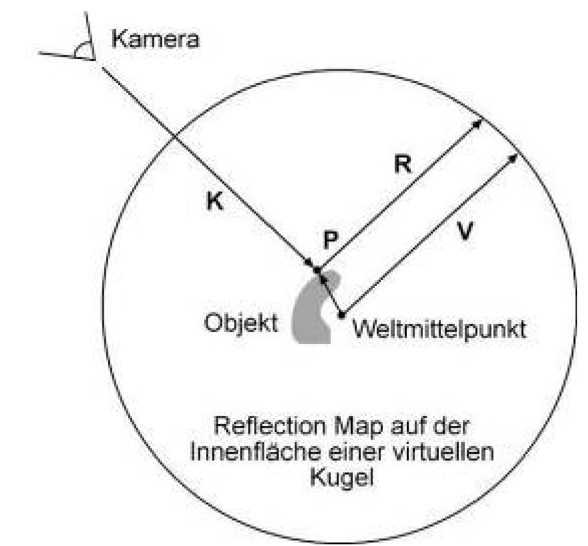


Abbildung 16 Grundidee von Reflection Mapping [2]

Dieses Verfahren ist relativ schnell und liefert auch gute Resultate. Dies sofern die Textur den Himmel oder ein weit entferntes Objekt repräsentiert.

Es gibt aber auch grosse Probleme, welche die Anwendungsmöglichkeiten stark einschränken:

- Es muss eine Reflection Map erzeugt werden.
- Die Reflektionsberechnung ist strenggenommen nur dann korrekt, wenn sich **P** im Weltmittelpunkt befindet.
- Kugelkoordinaten müssen über trigonometrische Funktionen berechnet werden.
- Es gibt keine Verdeckungen von Objekten.
- Objekte können sich nicht gegenseitig spiegeln.
- Es treten grosse aliasing Probleme auf. (Kap. 4.1.6)

4.1.5.6 Environment Mapping

Dieses Verfahren ist eine Weiterentwicklung des Reflection Mapping. Anstatt einer Kugel wird ein Würfel als Reflection Map verwendet. Damit können die aliasing Probleme stark reduziert werden. Es bleiben trotz der Verbesserungen immer noch einige Probleme offen:

- Es gibt keine Verdeckungen.
- Objekte können sich nicht gegenseitig spiegeln.

4.1.6 Filterung diskreter Texturen (Antialiasingproblem)

Die bisher beschriebenen Mapping Funktionen können einer Bildschirmkoordinate eine Texturkoordinate zuweisen. Wenn wir dies tun, dann treten grundsätzlich Aliasingeffekte auf. Ein gutes Beispiel dafür sieht man in der Abbildung 17.

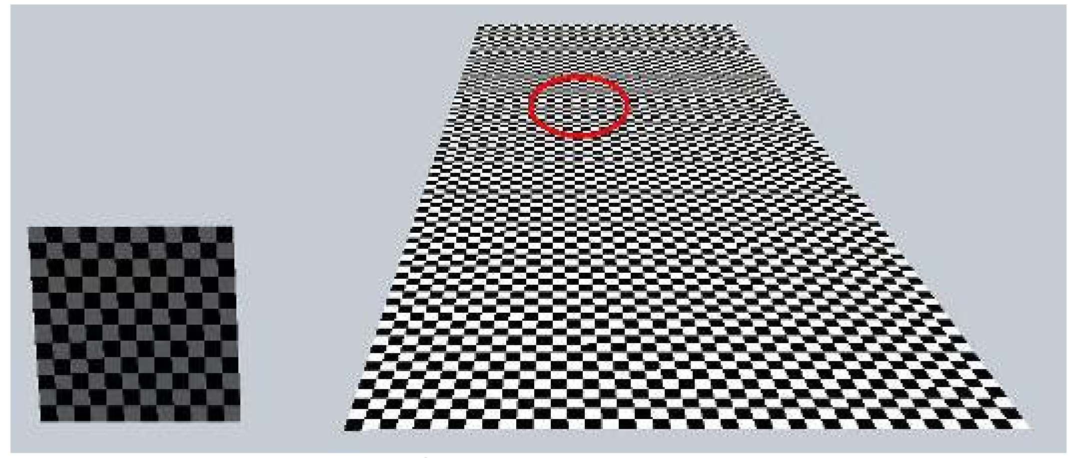


Abbildung 17 Scheinstrukturen im Bild (Aliasing)

Es fällt auf, dass im Vordergrund die Darstellung des Karos noch einigermassen funktioniert. Aber je weiter im Hintergrund, desto mehr Scheinstrukturen erscheinen.

Dieses Problem tritt auf, wenn eine Textur mit einer hohen Auflösung im Hintergrund dargestellt werden soll. Je weiter im Hintergrund, desto kleiner sollte eigentlich die Textur werden. Dies geschieht hier aber nicht. Es wird immer wieder versucht, die gleiche Textur mit der gleichen Auflösung, darzustellen.

Verfahren, welche diese Effekte minimieren, werden im Folgenden aufgezeigt:

4.1.6.1 Mip Mapping

Dies ist eines der bekanntesten Verfahren zur Verminderung des Aliasingeffektes (MIP = multum in parvum = Vieles im kleinen). Eine Mip-Map ist eine Folge von quadratischen Bildern, wobei jedes Bild um eine Zweierpotenz kleiner ist. Man spricht bei diesen Stufen auch von Level of Detail (LoD).

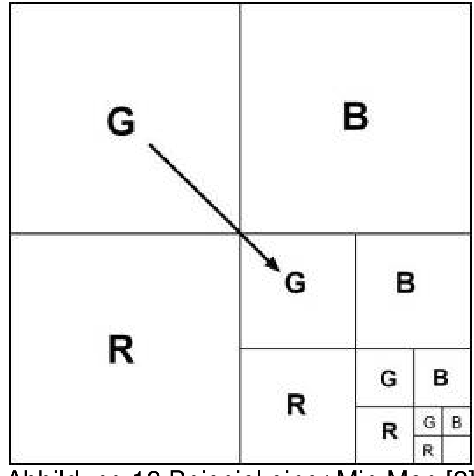


Abbildung 18 Beispiel einer Mip Map [2]

Mit dieser nur ein einziges Mal berechneten Vorfilterung können beliebige Flächenelemente, abhängig von der Betrachtungsdistanz, mit einer passenden Textur versehen werden. Es kann aber sein, dass keine passende Detailstufe vorhanden ist, welche zum Flächenelement passt. Es wird dann nicht nur bilinear innerhalb der nächsten passenden, sondern trilinear zu den nächsten beiden Stufen (also zur „zu guten“ und „zu schlechten“) interpoliert. Somit finden keine "Oversamplungen" der Bilddaten mehr statt und der Aliasingeffekt wird vermindert.
Mip- Map's haben aber auch negative Aspekte:

- So führen gerade die Sprünge zwischen den einzelnen LoD's zu Unstetigkeiten. Sie werden als Linien zwischen den MIP- Bändern sichtbar. Jedes dieser Bänder hat einen Schärfegrad, welcher sich deutlich vom anderen unterscheidet.

- Bei Stauchungen, welche keiner Zweierpotenz entsprechen, kommt es zu Verzerrungen und Schärfeverlusten.

4.1.6.2 Summed Area Tables

Da bei Mip-Mapping die einzelnen Detailstufen der Bilder quadratisch sein müssen, wurde ein ähnliches Verfahren entwickelt, bei dem die Bilder aber nicht quadratisch sein müssen. Dieses Verfahren nennt man Summed Area Tables.

4.1.6.3 Elliptical Weight Average Filter (EWA)

Auch der EWA Filter hat die gleiche Funktion wie Mip-Mapping oder Summed Area Tables. Nur wird hier die Form der LoD's noch komplexer gewählt, es wird eine Ellipse verwendet. Das Resultat wird durch diese Verwendung zwar besser, nur ist auch der Rechenaufwand dementsprechend höher. Dies hat zur Konsequenz, dass EWA-Filter für Echtzeit-anwendungen nur bedingt einsetzbar sind.

4.1.6.4 Footprint Assembly

Einen Kompromiss zwischen EWA und Mip-Mapping stellt dieses Verfahren dar. Dazu wird die Textur ähnlich wie beim MipMap-Verfahren zerlegt und gespeichert. Um den Texturwert letztendlich zu ermitteln, wird aber nicht nur ein MIP berücksichtigt.

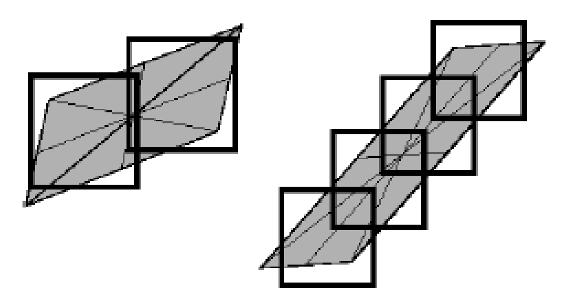


Abbildung 19 Beispiel von Footprint Assembly [2]

In der Praxis wird der Footprint durch das arithmetische Mittel über 2 Quadrate angenähert.

4.2 Untersuchungen und Resultate

Hier werden die einzelnen Untersuchungen vorgestellt und besprochen, welche zum Thema Texturen gemacht wurden.

4.2.1 Kameramapping

In Cinema 4D existiert eine Projektionsart, welche sich Kameramapping nennt. Unter Kameramapping ist das Mappen von Texturen auf ein Objekt, aus der Sicht einer Kamera zu verstehen.
Es soll hier gezeigt werden, ob sich diese Funktion eignet, um die Bilder der Nahbereichsphotogrammetrie auf ein Objekt zu mappen. Dabei ist eine Übernahme der Orientierungselemente so weit wie möglich anzustreben.

4.2.1.1 Übernahme der Orientierungswerte

Es hat sich gezeigt, dass alle Werte der äusseren Orientierung übernommen werden können. Zum Teil sind es die gleichen Werte, welche aber in Cinema 4D anders benennt sind. Dies bereitet zum Teil Probleme, da nur durch Austesten der Parameter diese richtig zugeordnet werden konnten.

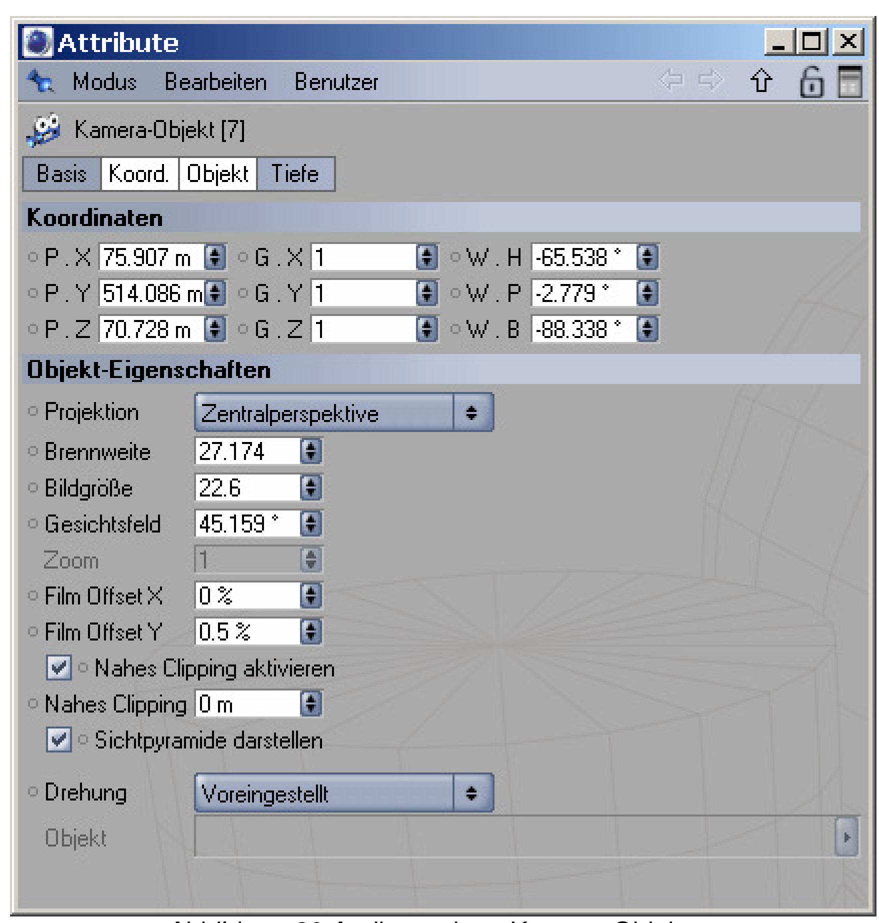


Abbildung 20 Attribute eines Kamera-Objektes

Koordinaten

Im ersten Abschnitt ist es möglich, die Koordinatenwerte und Raumwinkel der Kameraposition einzugeben. Die Raumwinkel müssen dabei aber im Euler-System vorliegen. (Kap. 4.2.1.2)

Objekt- Eigenschaften

Wenn man eine Kameraprojektion ausführen will, muss als Mapping-Projektion die Zentralperspektive gewählt werden.

Die Brennweite entspricht der, welcher üblicherweise in der Photogrammetrie verwendet wird. Sie kann somit genau aus der Kamerakalibrierung übernommen werden.

Bei der Bildgrösse handelt es sich eigentlich um die Grösse des "Dias" bei analogen Kameras. Da heute die Photogrammetrieprojekte aber meist mit einer Digitalkamera verwirklicht werden, entspricht diese Grösse der Grösse des optischen Chips der Kamera. Leider steht für die Chipgrösse kein kalibrierter Wert zur Verfügung. Somit muss der Hersteller-Wert genommen werden. Es zeigte sich, dass dieser meist nicht stimmt. Dies schlug sich in einem Massstab in der Textur nieder. Durch probieren konnte jedoch eine Näherung des Wertes bestimmt werden.

Die Filmoffsets sind eigentlich die Verschiebungen der Bildhauptpunkte in Prozenten in Bezug auf die Chipgrösse ausgedrückt. Wenn man die Verschiebungen der Bildhauptpunkte umrechnet, kann man sie auch direkt berücksichtigen. Dabei ist es wichtig, dass man sich immer auf die aktuelle Chipgrösse bezieht. Bei den Testversuchen gab es fast kein Bild, welches nicht mittels eines zusätzlichen Offsets noch an das Objekt angepasst werden musste. Besonders heikel sind Texturierungen bei Fenstern oder kleinen Öffnungen. Bei diesen Objekten stellt sich allerdings die Frage, ob diese Genauigkeit mit der Orientierungsgenauigkeit noch gegeben ist. Um dies zu veranschaulichen, wurden aus dem Projekt Wildenstein die Genauigkeiten von drei Bildern begutachtet. Die Standartabweichungen der Raumwinkel wurden mit dem mittleren Objektabstand in Abweichungen auf dem Objekt umgerechnet.

Bild Nr 1	**mittlerer Objektabstand**		**70.00 m**
Omega	0.1920 Gon	0.2111 m	
Phi	0.0680 Gon	0.0748 m	
Kappa	0.1330 Gon	0.1462 m	

Bild Nr 3	**mittlerer Objektabstand**		**70.00 m**
Omega	0.1400 Gon	0.1539 m	
Phi	0.0560 Gon	0.0616 m	
Kappa	0.0320 Gon	0.0352 m	

Bild Nr 6	**mittlerer Objektabstand**		**65.00 m**
Omega	0.2270 Gon	0.2318 m	
Phi	0.0710 Gon	0.0725 m	
Kappa	0.1910 Gon	0.1950 m	

Tabelle 2 Vergleich der Raumwinkelgenauigkeiten

Die Abweichungen betragen bis zu 23cm, sind aber auch meist über 5cm. Was für uns heisst: Eine manuelle Korrektur der Bilder ist unumgänglich ist. Für eine automatische Texturierung, in welcher die Parameter übernommen werden, ist eine genauere Orientierung nötig.

4.2.1.2 Umrechnung der Orientierungswinkel von PHIDIAS

Wie in Kapitel 4.2.1.1 schon erwähnt, benützt Cinema 4D das „eulerische“ Winkelsystem. Das Auswerteprogramm für die Nahbereichsphotogrammetrie (PHIDIAS) liefert die Winkel aber in einem „Standart“ System, welches in der Photogrammetrie üblich ist. Es besteht aber die Möglichkeit, diese Winkel mit folgenden Beziehungen umzurechnen.

Die "Standart" Rotationen welche in der Photogrammetrie gebraucht werden sind:

1. Omega, ω Drehung um die X-Achse
2. Phi, φ Drehung um Y-Achse
3. Kappa, κ Drehung um Z-Achse

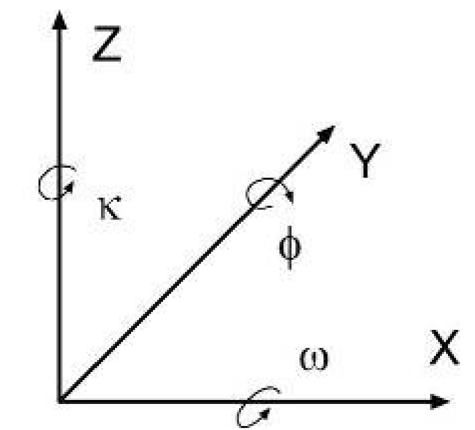


Abbildung 21 "standart" Koordinatensystem

Daraus ergibt sich die untenstehende Rotationsmatrize.

$$\begin{pmatrix} \cos(\varphi)\sin(\kappa) & \cos(\omega)\sin(\kappa)+\sin(\omega)\sin(\varphi)\cos(\kappa) & \sin(\omega)\sin(\kappa)-\cos(\omega)\sin(\varphi)\cos(\kappa) \\ -\cos(\varphi)\sin(\kappa) & \cos(\omega)\cos(\kappa)-\sin(\omega)\sin(\varphi)\sin(\kappa) & \sin(\omega)\cos(\kappa)+\cos(\omega)\sin(\varphi)\sin(\kappa) \\ \sin(\varphi) & -\sin(\omega)\cos(\varphi) & \cos(\omega)\cos(\varphi) \end{pmatrix}$$

Formel 1 Rotationsmatrize des "standart" Koordinatensystems [5]

Die Rotationen, welche in Cinema 4D verwendet werden, sind folgende:

1. Azimut, α Drehung um die Z-Achse
2. Elevation, ε Drehung um X-Achse
3. Roll, κ Drehung um Y-Achse

Dieses System ist analog zu jenem, welches man in der Flugnavigation verwendet. Dabei werden die Winkel aber anders benennt:

1. Heading, α Drehung um die Z-Achse
2. Pitch, ε Drehung um X-Achse
3. Bank, κ Drehung um Y-Achse

Diese Benennung wird auch in Cinema 4D eingesetzt.

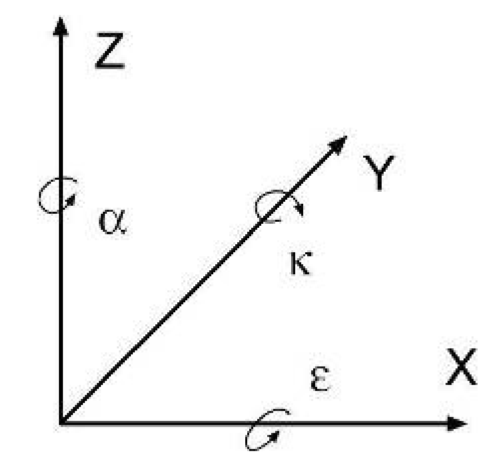


Abbildung 22 "eulerisches" Koordinatensystem

Daraus folgt wieder eine Rotationsmatrize:

$$\begin{pmatrix} \cos(\alpha)\cos(\kappa)+\sin(\alpha)\sin(\varepsilon)\sin(\kappa) & \sin(\alpha)\cos(\kappa)-\cos(\alpha)\sin(\varepsilon)\sin(\kappa) & \cos(\varepsilon)\sin(\kappa) \\ -\cos(\alpha)\sin(\kappa)+\sin(\alpha)\sin(\varepsilon)\sin(\kappa) & -\sin(\alpha)\sin(\kappa)-\cos(\alpha)\sin(\varepsilon)\cos(\kappa) & \cos(\varepsilon)\cos(\kappa) \\ \sin(\alpha)\cos(\varepsilon) & -\cos(\alpha)\cos(\varepsilon) & \sin(\varepsilon) \end{pmatrix}$$

Formel 2 Rotationsmatrize des "eulerischen" Koordinatensystems [5]

Aus diesen zwei Matrizen lässt sich relativ einfach der Zusammenhang zwischen den Winkeln herstellen. Da zum Beispiel an der Position (3,3) in beiden Matrizen derselbe Wert stehen muss. Somit können diese beiden Ausdrücke gleichgesetzt werden.

$$\cos(\omega)\cos(\varphi) = -\sin(\varepsilon)$$

Es folgt somit für alle Winkel:

$$\alpha = \arccos\left(\frac{-\sin(\omega)\cos(\varphi)}{-\cos(\varepsilon)}\right)$$

$$\varepsilon = -\sin(\cos(\omega)\cos(\varphi))$$

$$\kappa = \arcsin\left(\frac{\sin(\omega)\sin(\kappa) - \cos(\omega)\sin(\varphi)\cos(\kappa)}{\cos(\varepsilon)}\right)$$

Diese Formeln können nun z.B. in eine Exceltabelle integriert werden, welche ein einfaches und effizientes Umrechnen der Winkel ermöglicht.

Diese Umrechnungen können weggelassen werden, wenn man mit einer Auswertesoftware arbeitet, welche gerade die benötigten Raumwinkel berechnet. Ein Beispiel einer solchen Software wäre z.B. IWITNESS. [5]

4.2.1.3 Erstellung einer Mastertabelle

In der Folge der Bearbeitung des Projektes tauchte immer wieder das Problem auf, dass verschiedene Daten mühsam zusammengesucht werden mussten. Aus diesem Grund hat man sich zur Erstellung einer "Mastertabelle" entschieden, welche alle relevanten Daten zu einem Photogrammetrieprojekt enthält. Es wurden folgende Parameter in die Tabelle aufgenommen:

- allfälliger Shift für die gesamte Szene
- verwendete Bildgrösse (Chipgrösse)
- Verschiebungen des Hauptpunktes
- kalibrierte Brennweite

Sowie auch pro Bild:

- Name der Bilddatei
- Koordinaten mit angebrachtem Shift
- eulerischen Raumwinkel (Standartwinkel werden umgerechnet)
- Umgerechnete Verschiebung des Hauptpunktes

Das Ziel wäre, dass diese Tabelle von der Person, welche die Auswertung der Photogrammetrie macht, vollständig ausfüllt wird. Somit wird ein mühsames zusammensuchen der benötigten Parameter vermieden.

4.2.1.4 Problem der Verzeichnung bei den Bildern

Bei den meisten Auswerteprogrammen werden zusätzlich die Verzeichnungsparameter der verwendeten Kamera bestimmt. Da die Kameras meist nicht hochgenaue und stabile Objektive besitzen, kann die daraus folgende Bildverzerzeichnung relativ gross werden. Deshalb wurde ein Tool gesucht, mit welchem man auf einfache Weise diese Verzerrung auf den Bildern korrigieren kann.

Nach einer Internetrecherche und dem Studium der Anleitung von PHIDIAS kristallisierten sich zwei Lösungsansätze heraus:

- Korrektur der Verzeichnung in PHIDIAS
- Berücksichtigung der Verzeichnung mit dem Programm PTLens

PHIDIAS ist ein professionelles Nahbereichsphotogrammetrie Auswertungsprogramm. Es baut auf dem CAD-Programm Microstation auf und ist nur in Verbindung mit diesem einsetzbar.

PTLens ist ein Programm, welches ein Bestandteil der Panotools ist. Diese wurden ursprünglich zur Erzeugung von Panoramabildern entwickelt. Das Programm stammt aus dem Fotobereich und wurde auch für Fotozwecke zugeschnitten.

Anschliessend werden die einzelnen Vor- und Nachteile der zwei Programme dargestellt.

Vorteile PHIDIAS	**Nachteile PHIDIAS**
• Kalibrierungswerte können direkt übernommen werden • Korrekturmodell ist bekannt • Ist ein „messtechnisches" Programm • Programm wird bereits verwendet • Support gesichert	• Ist nicht einfach zu bedienen • Besitzt keine Vorschau • Kalibrierung der Kamera relativ aufwendig • Integriert in Microstation • PHIDIAS Projekt muss vorhanden sein
Vorteile PTLens	**Nachteile PTLens**
• Ist einfach bedienbar • Besitzt eine Bildvorschau • Kalibrierung mit einem Bild möglich • Ist Open source Software • Unabhängige „Stand alone" Software	• Kalibrierungswerte aus PHIDIAS können nicht übernommen werden • Es besteht ein neues Korrekturmodell • Programm stammt aus dem Fotobereich • Weiterentwicklung und Support nicht gesichert • Korrigiert nur lineare Effekte • Kalibrierungswerte für Objektive sind vorhanden, aber ohne Nachweis der Genauigkeit

Tabelle 3 Vergleich Phidias und PTLens

Es wurde beschlossen, dass die Korrektur der Bilder in PHIDIAS zu erfolgen habe. Anfänglich war ein Export der verzeichnungsfreien Bilder nicht möglich. Es wurde mit dem Support von PHIDIAS Kontakt aufgenommen und das Problem gelöst. In PHIDIAS können nun die Verzeichnungen der Bilder einzeln berechnet werden. Sie werden dabei als unkomprimierte TIF-Files ausgegeben.



Abbildung 23 Bild ohne Verzeichnugskorrektur



Abbildung 24 Bild mit Verzeichnugskorektur

Man kann die Bilder deutlich am schwarzen Rand des korrigierten Bildes unterscheiden.

4.2.2 Orthophoto in eine Szene integrieren

Gerade bei den Szenen, welche bis jetzt bearbeitet wurden, ist der Einsatz eines Orthophotos für die Texturierung der Umgebung sehr wichtig. Es soll aufgezeigt werden, ob man ein Orthophoto in eine Cinema 4D Szene integrieren kann.

Die Untersuchungen zeigten, dass ein Import eines Orthophotos unter gewissen Umständen möglich ist.
Will man ein Orthophoto auf ein Geländemodell legen, so müssen beide dieselbe geometrische Ausdehnung besitzen. Sonst wird das Orthophoto verzerrt.

Es ist zwar möglich, ein Orthophoto auf ein Geländemodell zu legen, nur wird die Szene dadurch relativ schnell gross und langsam. Zur Bearbeitung der Szene wäre es daher gut, zwei Szenen zu erstellen: Eine Szene mit dem Orthophoto und dem Höhenmodell und eine zusätzliche zur Bearbeitung des Objektes. Wenn diese beiden Szenen die gleiche Georeferenzierung besitzen, können sie anschliessend in Cinema 4D einfach zusammengefügt werden.

4.2.3 Anwendung von Displacement Mapping oder Bump-Mapping

In Kap. 4.1.5 werden verschiedene Arten des Mappings vorgestellt, welche den Körper optisch verformen. Jetzt stellt sich die Frage, ob solche Verformungen auch auf Fototexturen angewendet werden können.

Es wurde nun versucht, die vorhandenen Fototexturen mit den Methoden:

- Displacement Mapping
- Bump-Mapping

zu ergänzen, dass ein noch realistischer Effekt erzeugt wird.
Es zeigte sich, dass eine Verfeinerung der Texturen möglich war. Dies aber nur mit enormem Arbeitsaufwand. Dies folgte aus der Tatsache, dass für die Mappingmethoden ein Graustufenbild für die Darstellung des Höheneindruckes vorhanden sein muss. Es macht aber nur Sinn, solche Erhebungen z.B. bei Mauerfugen zu erzeugen. Wandelt man nun die Texturfotos in Graustufenbilder um, so kommt es auf glatten oder ebenen Flächen zu unerwünschten Vertiefungen oder Erhebungen, hervorgerufen durch feine Schattenwürfe oder Flecken auf den Objekten. Diese Fehler müssten dann wieder manuell entfernt werden.
Die Erzeugung eines Graustufenbildes für jedes einzelne Bild wäre somit zwar möglich, doch der Aufwand ist zu gross.

4.2.4 Darstellung von Level of Detail

Da eine vollständige Darstellung aller Objekte in einer 3D-Szene keinen Qualitätsgewinn bringt und die Performance unnötig belastet wird, ist die Lösung oft eine distanzabhängige Darstellung der Details.

In Cinema 4D wurde nach einer Möglichkeit gesucht, eine Szene mit LoD's darzustellen. In den Exportfunktionen wurde aber keine dementsprechende Funktion gefunden.

Da es aber bei einem VRML Export möglich ist, die Texturen einzeln abzuspeichern, könnte dieser eine Ansatzpunkt für eine ähnliche Funktion sein. Je nachdem was man für Qualitätsansprüche hat, kann man die Texturen in die gewünschte Qualität reasamplen. Das würde die Qualität zwar beeinträchtigen, aber die gesparte Datenmenge wäre enorm.
Dies entspricht nicht ganz der Idee von Level of Details, aber eine Reduktion der zu übermittelnden Datenmenge ist damit relativ einfach umsetzbar.

4.2.5 Radiometrische Korrekturen der Texturbilder

Meistens werden Bilder für die Texturierung aus Fotographien erstellt. Gerade wenn diese Fotos bei gutem Wetter gemacht wurden, sind an vielen Stellen Schattenwürfe sichtbar. Bei einer Verwendung der Bilder für die Texturierung sind solche Schatten oft unnatürlich und nicht erwünscht. Meist werden durch das verwendete Visualisierungsprogramm noch zusätzlich Schattenwürfe berechnet, was zu unschönen Mehrfachschatten führt.
Nebst dem Schattenwurf ändern sich je nach Betrachtungswinkel und Sonneneinstrahlung auch die Farben. Diese tritt gerade beim Übergang zwischen den einzelnen Seiten eines Gebäudes auf. Mappt man diese Fotos nun auf das Modell, so sind die Schnittstellen der einzelnen Fotos klar erkennbar.


Abbildung 25 Beispiel eines Texturbildes

Die beschriebene Problematik wird am Beispiel der Abbildung 25 sichtbar. Betrachtet man dieses Bild, so sieht man auf der linken Seite das Dach des Hauptgebäudes. Die Dachziegel haben in der frontalen Ebene eine viel hellere Farbe als auf der Seite.
Den Schattenwurf kann man beim Turm im Vordergrund gut betrachten. Der ganze Schatten fällt auf das Dach des Gebäudes. Mappt man dies nun auf ein Modell, so sieht das je nach Beleuchtung unrealistisch aus.

Es wurde nun nach Methoden und Abläufen gesucht, mit welchen diese unerwünschten Effekte aus den Bildern entfernt werden können.

Die zur Korrektur vorhandenen Programme wurden anschliessend gesichtet und anhand von Probedaten geprüft. Dabei handelte es sich um die folgenden Programme:

- **Microsoft Picture Manager**
  Mit ihm können Bilder angezeigt und automatisch korrigiert werden. Er wird standartmässig mit Microsoft-Office geliefert.

- **Mosaik Tool** (Erdas Imagine)
  Dabei handelt es sich um ein in Erdas Imagine integriertes Tool zum mosaikieren von Fernerkundungsdaten.

- **Image Equalizer** (Erdas Imagine)
  Dies ist ein in LPS (Leica Photogrammetry Suit) integriertes Tool zur radiometrischen Abgleichungen von Bildern.

- **Paint Shop Pro**
  Ist ein weit verbreitetes Bildbearbeitungsprogramm.

Die aus den Korrekturversuchen entstandenen Resultate wurden in einer Tabelle zusammengefasst.

Vorteile Microsoft Picture Manager	**Nachteile Microsoft Picture Manager**
• einfach Exportfunktion • einfache Bedienung • Standartprodukt (MS-Office)	• radiometrische Korrekturen nur über alle Kanäle gleichzeitig • radiometrische Korrekturen nur über ganzes Bild möglich • keine Batchfunktionen
Vorteile Mosaik Tool (Erdas Imagine)	**Nachteile Mosaik Tool** (Erdas Imagine)
• sehr ausgefeiltes Tool • Abgleich über alle Bilder möglich • viele Korrektur-Einstellungen möglich	• fügt alle Bilder zu einem zusammen • benötigt Erdas Imagine • radiometrische Korrekturen nur über ganzes Bild möglich
Vorteile Image Equalizer (Erdas Imagine)	**Nachteile Image Equalizer** (Erdas Imagine)
• Abgleich über alle Bilder oder einzeln möglich • einfach Exportfunktion • radiometrische Korrekturen über alle Kanäle oder je Kanal möglich • automatische Korrektur über mehrere Bilder möglich	• benötigt Erdas Imagine • nicht soviel Einstellungsmöglichkeiten • radiometrische Korrekturen nur über ganzes Bild möglich • Export nur als TIF oder IMG möglich
Vorteile Paint Shop Pro	**Nachteile Paint Shop Pro**
• radiometrische Korrekturen über alle Kanäle oder je Kanal • radiometrische Korrekturen über Ausschnitte im Bild möglich • Batch Export Funktion	• radiometrischer Abgleich nur einzeln möglich • keine Batchfunktion

Tabelle 4 Programmvergleich für radiometrische Korrekturen

Die Korrektur aller Fehler in den Bildern wäre, rein von den zur Verfügung stehenden Bearbeitungsprogrammen möglich. Wenn man aber alle Fehler in allen Bildern korrigieren will, ist dies mit einem enormen Arbeitsaufwand verbunden. Da meist die Zeit für solch eine intensive Korrektur fehlt, ist ein allgemeiner radiometrischer Abgleich über alle Bilder viel effektiver. Anschliessend an das Texturieren können grobe Fehler immer noch einzeln korrigiert werden. So kann ein Gleichgewicht zwischen Aufwand und Ertrag erzielt werden.

Für den radimetrischen Abgleich über alle Bilder stellte sich der Image Equalizer als effektivstes Programm heraus. Allerdings müssen die Bilder anschliessend mit der Exportfunktion von LPS umgewandelt werden, da der Image Equalizer ein TIF-Format schreibt, welches von den wenigsten Programmen gelesen werden kann. Gleichzeitig können die Bilder beim exportieren noch in ein JPEG-Format umgewandelt werden, was den Speicherbedarf der Texturen stark verringert.

4.3 Schlussfolgerungen

Die Untersuchungen haben gezeigt, dass eine Texturierung von komplexen 3D-Geoobjekten in Cinema 4D möglich ist. Es ist möglich, diese Objekte auf verschiedene Arten zu texturieren. Es können die Parameter der Nahbereichsphotogrammetrie übernommen werden, oder es kann ein manuelles Texturieren vorgenommen werden. Wobei ein Texturieren mit der Übernahme der Parameter zu bevorzugen ist, da ein manuelles "Aufkleben" der Texturen mehr mit probieren und versuchen zu tun hat als mit einem klar vorgegebenen Arbeitsablauf. Es besteht mit Sicherheit ein Vorteil zur bisherigen Vorgehensweise, bei der jedem einzelnen Polygon eines 3D-Geoobjektes eine Textur zugewiesen werden musste.

Eine Automatisierung des ganzen Texturierungsablaufes wäre denkbar. Die Erstellung der "Mastertabelle" könnte mit einem direkten Export in Cinema 4D um die Orientierungsparameter reduziert werden. Dies würde einen grossen Abtippaufwand ersparen und somit auch Fehlerquellen beseitigen. Eine denkbare Umsetzungsmöglichkeit wäre ein Plug-In in Cinema 4D, welches die Umrechnung der Winkel aus PHIDIAS und alle anderen nötigen Schritte vornimmt.

Um eine wirklich automatische Texturierung zu ermöglichen, ist eine Steigerung der Orientierungsgenauigkeit der Bilder unbedingt notwendig. Nur so kann eine nachträgliche Anpassung der Bilder vermieden werden. Und nur ohne diese Anpassungen macht eine Automatisierung wirklich Sinn.
Diese Steigerung der Genauigkeit beinhaltet auch die Bestimmung der exakten Grösse des verwendeten CCD-Chips der Digitalkamera. Die Vorgehensweise des Probierens, welche in den Untersuchungen angewendet wurde, stellt keine Lösung des Problems dar. Deshalb muss dieses Problem vor einer Automatisierung gelöst werden.

Momentan existiert kein "All in One" Softwarepaket. Es ist deshalb immer noch nötig, sich mit einer Unmenge an verschiedenen Softwaren herumzuschlagen. Schön wäre eine Software, welche von der Nahbereichsphotogrammetrie über Bildbearbeitung bis zur Texturierung alles kann. Leider ist eine solche Software in nächster Zeit nicht zu erwarten. Da der Markt in diesem Gebiet bis jetzt noch relativ klein ist, haben grosse Hersteller darin kein Interesse. Dies wird sich vielleicht in den nächsten Jahren ändern. Jedenfalls ist die Entwicklung im Auge zu behalten.

Betrachtet man die Ziele der Texturierung, so sind diese erfüllt worden. Es wurde ein Weg aufgezeigt, mit welchem die bisherigen Probleme gelöst werden können. Dies ist aber mit Sicherheit keine abschliessende Lösung. Eine Automatisierung der Abläufe ist unbedingt nötig, damit die Wirtschaftlichkeit solcher Projekte gesteigert werden kann.
Es wird in naher Zukunft noch einige Entwicklungen zu diesem Thema geben, welche aufmerksam verfolgt werden sollten.

5 Vegetation

Verfasser: Th. Schuler

Die realitätsnahe Darstellung von Vegetation in virtuellen Szenen ist bis heute nicht zufrieden stellend gelöst. Die hohe Komplexität der Objekte, welche in der Einzigartigkeit mit einem Mensch vergleichbar ist, stellt den Bearbeiter bereits bei der Modellierung vor fast unlösbare Probleme. Dazu kommen der enorme Speicheraufwand und die Renderungszeit.
Um eine möglichst realitätsgetreue Visualisierung der Realwelt zu erhalten, ist es aber vor allem bei Landschaftsabbildungen unverzichtbar, auch die Vegetation mit einzubeziehen.

5.1 Grundlagen (verschiedene Darstellungsformen)

Es gibt verschiedene Möglichkeiten, Vegetation und insbesondere Bäume darzustellen. In diesem Kapitel werden einige Varianten näher dargelegt.

5.1.1 Geometrisch vereinfachte Darstellung

3D-Symbolddarstellung

Eine Möglichkeit stellt die stark geometrisch vereinfachte Darstellung von Bäumen dar. Es ist eine abstrakte Darstellung, welche mehr auf das Vorhandensein von Vegetation hinweist als sie zu beschreiben.
Die Objekte sind aus einfachen Raumprimitiven wie Kegel, Kugel, und Zylinder zusammengesetzt. Oft besitzen die Objekte ein Glanzlicht, gekennzeichnet durch eine erhellte Fläche, welche bis zur Farbe Weiss gehen kann. Dies entspricht in keiner Weise der Realität.
Wenn eine einfach Visualisierungen mit einer gewollten gestalterischen Entfernung von der Realität erstellt werden will ist eine Symboldarstellung ein geeignetes Mittel. Denn eine Szene sollte immer homogen wirken, den Zusammenhang der verschiedenen Objekte gewährleisten und das im Vordergrund stehende hervorheben.
Eine Landschaftsszene mit Symbolen ist allerdings mehr eine Verfälschung der Wirklichkeit als ein Hilfsmittel zur Visualisierung. Symbolbäume anzuwenden um Wälder zu modellieren verzieht das Augenmerk vom Hauptobjekt auf Nebensächlicheres.

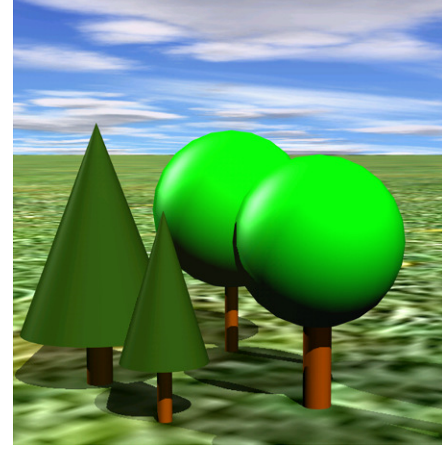


Abbildung 26 Symbolbäume

Rotationskörper

In den bisherigen Projekten der FHBB wurden die Bäume in MicroStation modelliert. Dabei wurden die Bäume als Zellen ins Modell eingefügt. Ein Baum besteht aus einzelnen Flächen. Für die Texturierung wurden teiltransparente Materialen verwendet. Jede Baumart wurde anschliessend als Zelle gespeichert.
Auch hier wurde eine vereinfachte Darstellung gewählt. So ist bei gewissen Ansichten oder genauerem Hinschauen die Zweidimensionalität der Objekte nicht zu verbergen. Zudem ist die Vielfältigkeit und Eigenartigkeit eines Baumes nicht gegeben. Hingegen ist bei beiden beschriebenen Arten ein grosser Anstieg der Datenmenge nicht gegeben.

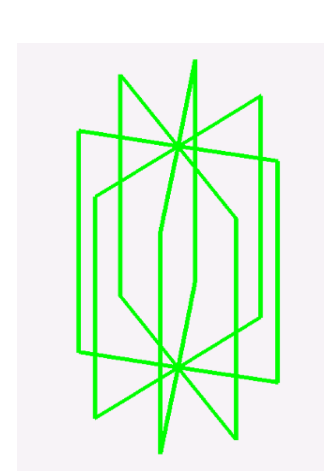
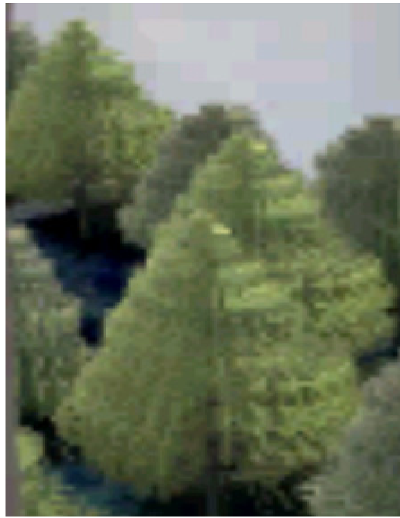




Abbildung 27 Rotationskörper in MicroStation und gerendert (aus Projekt Schloss Ebenrein und Schloss Zwingen 2001, FHBB)

Billboard (2D-Bilder)

Für eine einfache Visualisierungsaufgabe genügt oft ein zweidimensional digitales Photo als Fläche, welches immer zum Betrachter zugewendet wird. „Wenn allerdings Bildfolgen (Animationen) erstellt werden, so ist eine durchgehende dreidimensionale Modellierung aller Inhalte einer Landschafsszene erforderlich [16]“. Dies vor allem wenn Ansichten aus der Vogelperspektive erstellt werden, denn durch die Schräglage der Fläche erscheint das Bild flach auf der Oberfläche liegend. Diese Methode ist sehr gut geeignet für Visualisierungen auf mittlerer Massstabsebene, da dort eine grosse Zahl an Pflanzenobjekten mit relativ geringer Auflösung modelliert wird. [24]

5.1.2 Automatische Pflanzengenerierung

Eine interaktive dreidimensionale Modellierung von Pflanzen ist aufgrund der enormen Zahl an Einzelelementen, die aneinandergefügt werden müssen, praktisch nicht durchführbar. Andererseits folgt das Pflanzenwachstum relativ einfachen Gesetzmässigkeiten, aus denen Algorithmen für eine automatische Generierung der geometrischen Struktur abgeleitet werden können (www.sbg.ac.at/geo/agit/papers96/muhar.html) [16]

Einfach fraktale Graphiken

Man spricht von einer fraktalen Darstellung, wenn ein Objekt aus mehreren verkleinerten Kopien seiner selbst besteht. Bei einem Baum kommen fraktale Strukturen ohne strenge, sondern mit statischer Selbstähnlichkeit zur Anwendung, die sehr ähnlich wie reale Pflanzen aussehen. „Bei näherer Betrachtung zeigt sich jedoch, dass mit solch einfachen Beschreibungen viele Aspekte des Pflanzenwachstums missachtet werden [16]“. So ist zum Beispiel das Seitenverhältnis von kleineren Strukturen bei Pflanzen nicht immer gleich zur übergeordneten.

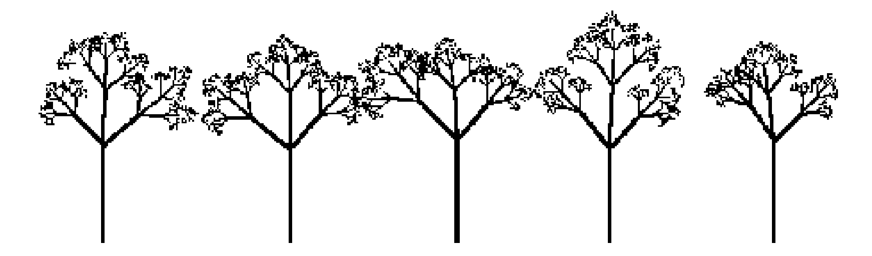


Abbildung 28 Anwendung statischer Variationen zur Generierung unterschiedlicher fraktaler Baumsymbole [16]

L-System

„Der dänische Botaniker Aristid Lindenmayer entwickelte eine formale Sprache, mit deren Hilfe das Wachstum einer Pflanze von der ersten Knospe weg so beschrieben werden kann, wie es sich auch in der Natur abspielt (Triebbildung, Anlage von Seitenknospen, Triebstreckung, Blattbildung, Ausbildung von Blütenorganen ab einem gewissen Alter etc.)[16].“
Dieser Ansatz sollte allerdings nur bei Sträuchern oder Ähnlichem eingesetzt werden, da das Wachstum bei Bäumen über Jahrzehnte nicht nur von den genetischen Informationen bestimmt wird, sonder auch die physischen Einflüsse eine wichtige Rolle spielen. [16]

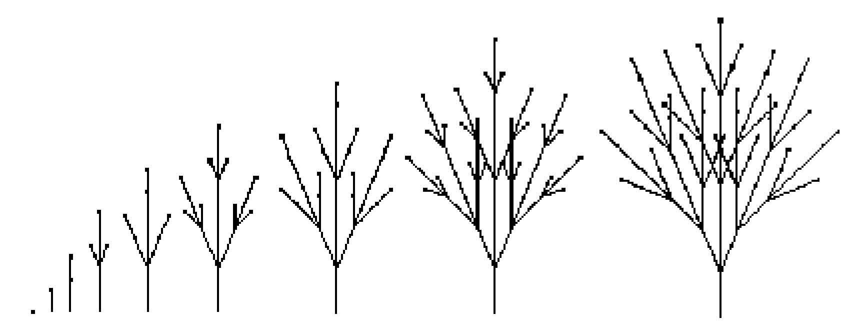


Abbildung 29 Beschreibung eines einfachen Wachstumsprozesses mit Hilfe der L-Grammatik

Prozedurale Methode am Beispiel des System AMAP (Hersteller: JMG-Graphics)

(Jäger und deReffye, 1992) entwickelten ein Verfahrensmodell, bei dem der Benutzer das Erzeugen der Triebe durch einige Parameter einer spezifischen Baumart steuern kann. „Dabei steht im Vordergrund die artspezifische Wahrscheinlichkeit des Auftretens bestimmter Eigenschaften eines Triebes, also die Art der Verzweigung, der Verweigungswinkel, die Dicke etc., jeweils in Abhängigkeit vom Alter der Pflanze[16].“ [18] / [24]

Schlussfolgerung

Eine Gegenüberstellung der beiden zuletzt erwähnten Methoden deckt sowohl Vor- als auch Nachteile der jeweiligen Methoden auf.
Die L-Systeme, bei denen eine formale Grammatik als Regelsystem dient, welches von einem Startzustand ausgehend einen Text erzeugt, der anschliessend graphisch interpretiert wird, sind meist sehr mächtig und kompakt, aber auch sehr wenig intuitiv.
Die prozedurale Methode verwendet spezielle parametrisierbare Generierungsalgorithmen, welche die Geometrie direkt erzeugt. Diese Methode ist intuitiv zu parametrisieren, dafür eingeschränkt zu verwenden (meist nur für eine Pflanzen Art).
Es gibt nun Programme, zum Beispiel das xfrog-Modelliersystem, welche die Vorteile beider Erzeugungsarten verbindet. Damit ist eine regelbasierte Objekterzeugung möglich (www.cgmi.inf.uni-konstanz.de/research/plants.html) [18].
Automatisches Modellieren wird vor allem eingesetzt, wenn die Wirkung des Vegetationswachstums veranschaulicht oder untersucht werden will.

5.1.3 Baum-Rekonstruktion aus Laserscanningdaten

Automatische Rekonstruktion von Einzelbäumen aus terrestrischen Laserscanning-daten (aus dem Bericht von Norbert Pfeifer, Ben Gorte und Daniel Winterhalder)[21]

Es wird eine Lösung aufgezeigt, wie man von einer Einzelpunktwolke eines Baumes, welcher mit Vorteil im Winter mit einem Laserscanner von allen Seiten aufgenommen wurde, über ein Rekonstruktionsprozess zu einem aus Zylinder bestehenden Modell kommt. Dabei wird auch jeder Ast miteinbezogen. Hierbei wird die Idee aufgegriffen, dass im Grunde ein Stamm und die Äste eines Baumes durch aneinanderreihende kreisförmige Zylinder beschrieben werden kann. Dabei wird einerseits ein Algorithmus für die Zylinderfolgen als auch einer für die Zylinderanpassung verwendet.
Der Algorithmus des Zylinderfolgens erfordert einen Ausgangspunkt in der Punktwolke, der ungefähr auf der Oberfläche der einzelnen Verzweigungen liegt, sowie einen ungefähren Wert für die Richtung der Zylindermittellinie. Der Algorithmus für das Zylindereinpassen erfordert eine gewisse Anzahl von Punkten, welche auf dem Zylinder liegen sollen.
Abgesehen vom manuellen segmentieren der Äste aus der Punktwolke und deren Auswahl relevanter Äste, können die oben genannten Ausgangswerte der Algorithmen auch automatisch erfolgt werden. (Gorte und Pfeifer, 2004)[22] beschreiben in „Strukturierte Laserscanningbäume unter Verwendung von mathematischer 3D Morphologie“ eine Methode für das Umwandeln der Punktwolke in einen voxel Raum oder 3D-Pixelraum genannt. Dabei wird der leere Raum innerhalb eines Baumes geschlossen (gefüllt). Zusätzlich wird für diesen Baum eine Skelettierung durchgeführt. Anhand des voxel-Baumes und der entstanden Skelettierung können die zukünftigen Zylinder ermittelt werden. Somit kann der Baum segmentiert werden.



Abbildung 30 links: Skelettierung, rechts: Voxel-Raum [22]

Die Untersuchungen ergaben erstaunliche Resultate. So konnte zum Beispiel bei einer Eiche automatisch 36 Äste gefunden werden. Die insgesamt 90 Zylinder sind mit einer durchschnittlichen Einpassungsgenauigkeit von 1.8cm bestimmt worden. Es konnten allerdings nicht alle Äste aufgebaut werden. Aufgrund der Tatsache, dass in Baumkronen immer weniger Punkte erfasst werden können, überrascht dieses Ergebnis allerdings nicht.



Abbildung 31 Das grosse Bild zeigt die automatisch erzeugten Äste. Im eingesetzten Bild erscheint das Resultat der Segmentation im voxel Raum. [21]

3D-Rekonstruktion von Strassen und Bäume für eine Stadtmodellierung (aus dem Bericht von George Vosselman)[23]

Es wird untersucht, wie ein realitätsnahes 3D-Stadtmodell erstellt werden kann. Dabei kommen Laserscanningdaten aus der Luft sowie 2D-Katastervermessungsdaten zum Einsatz. Zur Positionsbestimmung und Extraktion der Baumkronen werden die Farbeninformationen von Infrarotbildern verwendet. Die Stammposition der einzelnen Bäume liegt über dem Nullpunkt einer Skelettierung der ermittelten Vegetationsflächen. Die Vegetationsflächen werden durch den ISODATA (Iterative Self-Organizing Data Analysis Technique) Klassifikationsansatz ermittelt.
Mit dieser Methode wurden 177 von 182 Bäumen in einer Strasse richtig erkannt. Die Störungen betrafen unter anderem Mehrfachpositionierungen sowie kleine Gebäude oder Strassenlaternen, welche als Bäume klassifiziert wurden.
Die Bäume werden dargestellt durch Formmodelle, welche die Höhe als Funktion der Breite umsetzt.

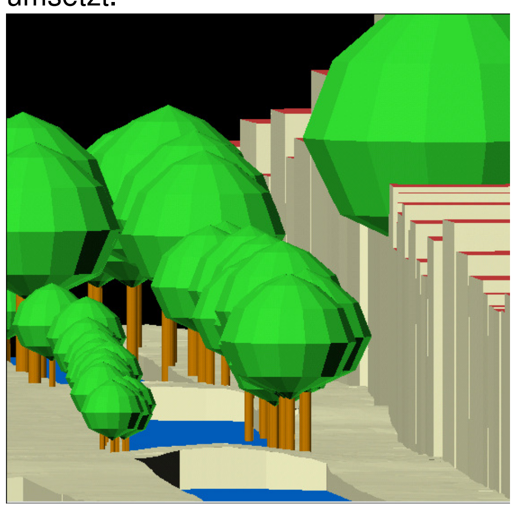


Abbildung 32 3D Stadtmodell mit Bäumen, Brücken, Strassen, Gebäuden [23]

5.1.4 Schlussfolgerungen

Es gibt verschiedene Arten wie Vegetation und insbesondere Bäume die Visualisierung einer 3D-Szene unterstützen können. So haben die geometrisch vereinfachten Darstellungen mehr einen unterstützenden Charakter um die Vegetation in einer Szene anzudeuten oder darauf hinzuweisen. Denn der Betrachter wird darauf aufmerksam gemacht, dass die Umgebung lebt und eine Abbildung der Umwelt mit all ihren Eigenschaften sein soll. Doch gleichzeitig wir ihm Bewusst gemacht, dass es ein abstraktes Modell ist und bleibt.
Eine Rekonstruktion aus Laserscanningdaten ist mehr für die Forstwirtschaft oder wissenschaftlichen Untersuchen über den Zustand der Wälder gedacht. Unter dem Aspekt einer realitätsnahen Visualisierung, sind die erzielten Resultate im Gebiet der vereinfachten Darstellungen anzusiedeln. Sicherlich ist jede Abbildung von Bäumen eigentlich eine vereinfachte Verwirklichung der Natur, doch mit dem visuellen Effekt welche eine Baummodellierung aus einer Vegetationsbibliothek (Kapitel 5.2.2) erzielt, kann die automatische Rekonstruktion aus Laserscanningdaten nicht mithalten. Zudem ist müsste in zukünftigen Projekten Laserscannpositionen im Wald miteinbezogen werden. Diese Methode ist zeitaufwendig und nicht einzusetzen bei einer Abbildung die den einzigen Zweck verfolgt die Umgebung in einem realen Bild ablichten zu lassen ohne grossen Zwang auf die geometrische Genauigkeit.

5.2 Untersuchungen und Resultate

5.2.1 Modellierung aus Laserscanningdaten

Da wir wie im Fall Homburg die gesamte Szene mit einem Laserscanner aufnahmen und so eigentlich gratis noch dazu eine Baumpunktwolke erhalten haben, ist es die Idee aus diesen bis anhin wertlosen Daten Bäume zu modellieren.
Der erst Schritt besteht darin in Cyclone aus der Punktwolke zuerst ein einzelnes Punktobjekt Baum zu generieren. Diese neue Punktwolke als ASCII-File herauslesen und in Cinema 4D einzufügen. Leider merkt Cinema 4D (im Gegensatz zum DXF-File Import) bei Punktimporten in den Strukturmanager nicht, dass die Y/Z-Achsen vertauscht sind und somit muss vorgängig in einem Texteditierprogramm dies korrigiert werden. Ebenso muss das vorgegebene Datenformat strengstens eingehalten werden. Die Werte pro Zeile müssen mit einem Tabulator, Komma, Semikolon oder einer Kombinaton daraus getrennt sein. Das Ende jeder Zeile muss mit dem ASCII-Zeichen für LF (Line Feed) oder dem Zeichen CR (Carriage Return) enden. Zudem kann kein Material an Punkte direkt vergeben werden, damit man den Baum grün einfärben könnte.

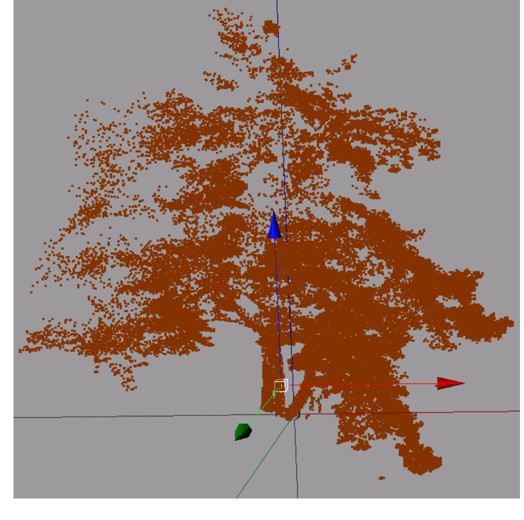


Abbildung 33 Punktmodus in Cinema 4d

Aus der Fülle von Möglichkeiten die Cinema 4D allgemein bietet werden einige als interessant zur Modellierung von Bäumen betrachtet und näher beschrieben.

Partikel

Die Partikeldefinition im Handbuch lautet: "Ein so genannter Emitter spuckt einen Strom von Teilchen(Partikel) aus. Das zeitliche und räumliche Verhalten dieser Partikel werden durch zahlreiche Parameter und Modifikatoren gesteuert.“ [6]

Bereits anhand dieser Definition kann erahnt werden, dass Punkte wenig mit Partikel gemeinsam haben. Denn Partikel werden erzeugt und bewegen sich im Raum, bestehen also nicht schon.
Partikelsysteme und „Thinking Particels“ werden für die Erzeugung von bewegenden Objekten wie Fischschwärme, Feuer- und Raucheffekte verwendet [7]. Wir haben aber bereits bestehende statische Punkte. Ein Baum hat also mit Partikel nicht viel zu tun.

Metaball

Eine weitere Möglichkeit stellt das Modelling-Objekt „Metaball“ dar. Dabei wird eine elastische Hülle, um Kugeln, Splines und Punkte gelegt. „Sobald eines dieser Objekte in das Metaball-Objekt im Objekt-Manager hineingeworfen wird, wird eine Hülle dadurch aufgeblasen [6].“Die wichtigste Einstellung welche bei einem Metaball gewählt werden kann, ist die Hülle. Mit dieser Einstellung bestimmt man, wie eng die Hülle am Objekt anliegen soll. Höhere Werte bedeuten, die Hülle wird enger um die Objekte gelegt. Der obere Grenzwert liegt bei 1000 [6]. Dies ist allerdings immer noch viel zu niedrig für eine Punktwolke mit einer Punktdichte von ca. 5cm. Die Hülle wird so gross, das die Dimension eines Baumes nicht mehr mit der Realität übereinstimmen. Die Performance leidet extrem unter der grossen Anzahl von Punkten. Bei 85'000 Punkten (durchschnittlich gut aufgenommener Baum) ist dies das letzte was man an diesem Tag mit Cinema 4D macht.

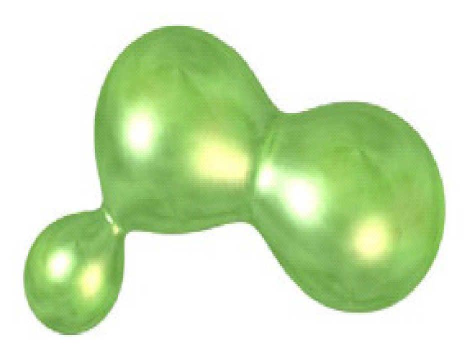


Abbildung 34 Beispiel eins Metaball-Objektes [6]

Atom-Array

Bei einem Atom-Array werden alle Objektkanten durch Zylinder und alle Objektpunkte durch Kugeln ersetzt. Wir haben keine Polygone in unserem Objekt, sondern nur Punkte, also werden auch keine Zylinder dargestellt

Dieses Modelling-Objekt ist besser als ein Metaball da die entstehenden Kugeln kleiner gewählt werden können und somit Grösse / Form des Baumes einigermassen eingehalten werden kann. Auch hier gibt es ein Performance Problem, allerdings nicht derart gross wie bei Metaball. Es ist wichtig, dass man genügend Punkte über den ganzen Baum zur Verfügung hat, damit keine Löcher entstehen und damit die Kugelformen nicht sofort ersichtlich sind. Performance und Punktdichte sind also im Widerspruch zueinander. Generell kann gesagt werden, dass diese Methode hochineffizient genau ist.



Abbildung 35 Beispiele von Atom-Array-Bäume

Plug-In

Die Firma motion gimmick – MediaArt (www.motion-gimmick.net) bietet ein Plug-In namens Patch_It für 20 Euro an. Mit diesem Programm können über selektierte Punkte Triangulationen (Polygonobjekte) automatisch erzeugt werden. Bei einfachen geometrischen Objekten ist das Auswählen von Punkten nicht schwierig. Bei einer Punktwolke könnte dies allerdings etwas schwieriger werden. Eine manuelle Nachbearbeitung wäre sicher nötig. Es wäre eine Interessante Alternative.

Manuelle Polygonerzeugung

Zum Schluss wird das eigenhändige Modellieren untersucht, also Polygone über die Punktwolke zu ziehen. Dabei wird immer wieder ein falscher Punkt angewählt und so entstanden ungewollte Polygone. Um dies zu verhindern, müsste man bereits die Punktwolke in Cyclone in hunderte unterteilen (am besten wäre in Stamm / Äste und Blätter). Was die Polygonkonstruktion in Cinema 4D einigermassen vereinfachen würde. Anderseits, ist der Aufwand aus einer riesigen Punktwolke nur schon einen einzelnen Baum zu extrahieren nicht zu unterschätzen. Ein gewisses Gespür und ein gutes räumliches Sehvermögen sind dabei unabdingbar, um nur unnötige Punkte zu löschen.

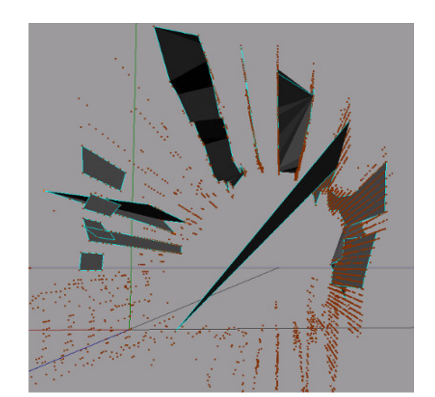


Abbildung 36 Beispiel falscher Vermaschung

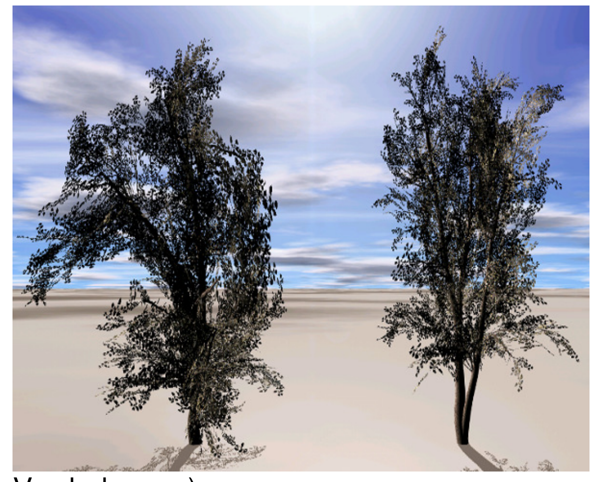
Diese Variante ist zu verwerfen. Bereits die Navigation in der Punktwolke, um Punkte auszuwählen, ist derart schwierig und mühsam, dass man fasst verzweifelt. Um einen gesamten Baum auf diese Weise zu modellieren braucht man ungefähr 3Tage, was bereits aus wirtschaftlichen Überlegungen keinen Sinn macht.

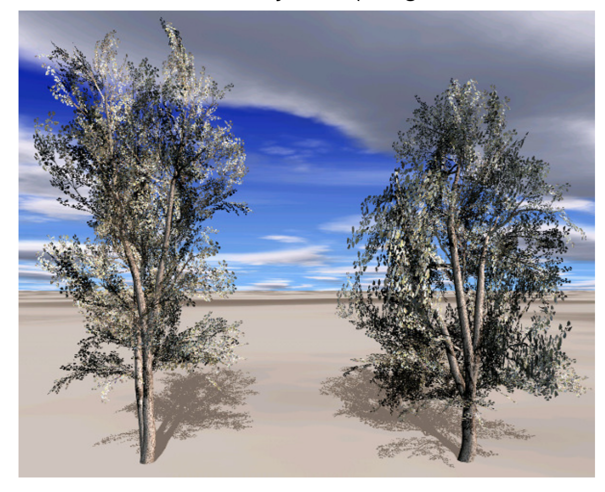
Anpassungen an gegebene Objekte

Eine einfache und sehr effektive Art ist nur die Höhe, Breite und Tiefe in Cyclone über Tools/Measure/Distance/Point to Point zu messen und so in Cinema 4D den Baum aus der Bibliothek anzupassen. Mehr dazu im Kapitel 5.2.3.

5.2.2 Vegetationsbibliothek (vbexteriors$^{\text{3D Plants}}$)

Vbexteriors für Cinema 4D ist als Plug-In von der Firma VBvisual.com aus Düsseldorf erhältlich und beinhaltet über 200 fotorealistische 3D Pflanzen. Die zur Verfügung gestellten Pflanzen sind unterteilt in Bäume, Sträucher und Büsche sowie Blumen. Bäume und Büsche sind sogar in zwei verschiedenen Detaillevels vorhanden. Da Material und Mapping bereits zugewiesen sind, ist das bedienen der Objekte sehr einfach. Durch den geringen Datenaufwand und die optimierte Materialien ist die Renderzeit schnell, was besonders wichtig für Animationen ist.
Bäume können sehr schnell und einfach in einer Szene eingefügt werden. Ebenso einfach ist es sie in ihrer Gesamtheit in der Höhe, Breite, Tiefe und Orientierung zu verändern. Dies kann mit den grundlegendsten Befehlen von Cinema 4D gelöst werden (Skalierung, Drehung, Deformation).
Die Äste sind in mehrere Polygone unterteilt. Will heissen, dass ein Ast aus mehreren Polygonen zusammengesetzt wird, die sich in allen Ebenen überschneiden sowie eine andere Geometrie aufweisen. Dies ruft einige Schwierigkeiten hervor, wenn nur ein einzelner Ast verändert werden will, da man bei gewissen Baumarten bis zu dreimal den gleichen Vorgang wiederholen muss. Wenn allerdings auf einem bestimmten Teil des Baumes Polygone wild selektiert werden um sie zu verformen, erhält man innert nützlicher Zeit ansprechbare Resultate welche mit dem Ausgangsobjekt nicht mehr viel gemeinsam haben. Dabei muss allerdings darauf geachtet werden, dass eine zu grosse Verformung einzelner Dreiecke vermieden wird, da es sonst beim rendern auffällt. Noch grundsätzlichere Veränderungen werden erzielt durch so genannte Deformationsobjekte (Biegen, Stauchen,





Verdrehen,…).

Abbildung 37 Beispiele von Veränderungen an einem Baum von vorne und hinten; im linken Bild ist der Originalbaum rechts, im rechten Bild umgekehrt

Einzelne neue Äste hinzu modellieren geht nur über mühsames Kopieren der vorgegebenen eingefrorenen Selektionen des gesamten Objektes Baum, da die Zuweisung zwischen Material und Mapping nicht aufgesplittert werden darf. Dazu muss zuerst der gesamte Baum kopiert werden. Dann den ausgewählten Ast in allen eingefrorenen Selektionen auswählen und den Rest löschen. Man hat nun nur noch den ausgewählten Ast als ein einzelnes Objekt welches irgendwo platziert werden kann. Dieser Prozess ist mit enormem Zeitaufwand verbunden.

Die ungefähre Höhe etc. in Cyclone zu messen ist einfach und erzielt den Effekt, damit am Schluss in Cinema 4D zumindest die „äusseren“ Dimensionen eines Baumes stimmen. Die Lage kann selbstverständlich auch aus Cyclone herausgelesen werden. Dabei müssen die Koordinatenwerte aufgeschrieben werden sowie der Koordinatenoffsets der „Cinema 4D-Szene“ abgezogen werden. In Cyclone sind die Koordinatenwerte am einfachsten herauszulesen, wenn ein der gewünschte Punkt selektiert wird und dann über Tools / Info / Objekt Info abgerufen wird.



Abbildung 38 Beispiel eines selbst erstellten Astes

Schlussfolgerung

Ein sehr nützliches und hilfreiches Plug-In. Vielleicht müsste man sich überlegen selber abgeänderte Bäume abzuspeichern, damit sie wieder benützt werden können. Denn im Wald selber haben Bäume oft nur an der Krone Äste, am Stamm selten oder zumindest sehr gering in Anzahl und Ausdehnung. Dadurch könnte auch der grossen Datenmenge etwas Einhalt geboten werden.
Was ich auf jeden Fall empfehle, ist bei der Erfassung des gesamt Objekts in den Workshops einen Baumkataster anzulegen (wo welche Baumsorten sind), um der Realität noch einen Schritt näher zu kommen.

5.2.3 Kombination von vbexteriors$^{3D\ Plants}$ und Laserscanningdaten

Die Gesamtpunktwolke in einzelne Bäume aufzusplittern ist nicht effizient. Die Ausdehnungen eines einzelnen Baumes genau zu bestimmen ist schwierig. Deshalb werden hier grössere Gebiete(ca. 30X50m) aus der Punktwolke ausgeschnitten und in Cinema4D eingefügt (Ablauf siehe Anfang des Kapitels 5.2.1). Es kann schell mal vorkommen, dass ein File mit 400'000 Punkte eingelesen wird. Die Leistungsgrenze von Cinema 4D ist dabei an der Grenze angelangt und etwas Geduld muss dabei vorausgesetzt werden. Ein Bibliotheksbaum wird nun durch graphische Kontrolle an den Laserscanningdaten angepasst.

In Homburg ist das DGM in der näheren Umgebung aus dem 25m Raster nie so genau wie aufgenommene Laserscanningpunkte am Boden. Vor allem in Bezug auf die Höhe. Die Punktwolke ist bis zu 2m über dem DGM-Grund. Die Höhe des Baumes kann somit nicht genau bestimmt werden. Um einen Baum an die richtige Lage zu bringen, ist es am einfachsten die Koordinaten eines Stammpunktes aus der Punktwolke anzeigen zu lassen und den Baum an diese Stelle zu verschieben. Die exakte Form aus einem Dickicht von Punkten zu erkennen ist eine Meisterleistung. Zudem muss eine Änderung an der Ausdehnung immer wieder über das Einschalten der Punktwolke kontrolliert werden.

Das einlesen der Punktwolke ist Zeit- und Nervenaufwendig. Es werden auch bei dieser Methode aufgeschriebene Koordinaten für die Positionierung verwendet. Der gleiche Vorgang wird einfach erst in Cinema 4D erledigt. Wenigstens müssen so die Koordinatenwerte nicht noch zusätzlich umgerechnet werden. Die Form eines Baumes mit hinterlegter Punktwolke anzupassen, also visuell, ist sehr schwierig. Hingegen kann die Abweichung aus der Senkrechten relativ gut bestimmt werden. Diese Variante bringt keine Steigerung der Produktivität gegenüber einer Koordinatenabfrage in Cyclone. Es sind beide Variante etwa gleichwertig, ausser dass bei der Übernahme der Punktwolke in Cinema 4D die Bäume durch geschicktes entfernen von Punkten in der Umgebung bereits erstellter Bäume und gleichzeitigem Ausschalten dieser Bäume eine gute Anpassung der Form vorgenommen werden kann.
Es muss hier noch erwähnt werden, dass die Grundlagedaten von Homburg nicht überall für diese Aufgabe geeignet waren. Da die Laserscanniggruppe, welche die Aufnahmen in Homburg erledigten, unter Zeitdruck standen, haben sie selten einen 360grad Scan durchgeführt, sondern nur diejenigen Bereiche erfasst welche sie interessierten. Dadurch sind lückenhafte Aufnahmen vom Wald entstanden. Zudem Stand der Scanner oft in Augenhöhe der Baumkrone, so dass die Stämme fehlen. Es ist in Zukunft bereits bei der Planung darauf zu achten, was man mit den erfassenden Daten anfangen will und dementsprechend im Vorrau richtig handelt. So sollte darauf geachtet werden, dass der Waldrand in der nächsten Umgebung des Objektes mit Laserscanner erfasst wird. Noch besser wären paar gut platzierte Standorte im Wald.

5.2.4 Wiese

Basic Material von Cinema 4D

Zur Thematik wie eine schöne saftige Wiese modelliert werden kann, stellt Cinema 4D keine spezielle Lösung zur Verfügung. In Cinema 4D gibt es ein Material, welches als Textur ein Bild einer saftigen Kleewiese verwendet. Wenn dieses Material dem DGM vergeben wird erhält man in gewissen Zoomeinstellungen tatsächlich den Eindruck als wäre es eine Wiese. Leider ist dies nicht bei jeder Entfernung vom Gelände der Fall. In Bodennähe erkennt man sofort, dass es sich nicht um ein 3D-Objekt handelt. Zudem sieht es einfach nicht zufrieden stellend aus. Eigenversuche und vorgenommene Änderungen fruchteten ebenfalls nicht.

Spezifische Wiesentextur

VBvisual.com hat neben dem Plug-In von Bäumen ein weiteres über 2D-texturierter Wiesen im Angebot welche einen gewissen 3D Eindruck vermitteln.

5.2.4.1 Hair Department

Die BGS-Group (www.bgs-group.de) hat ein Plug-In für Cinema 4D entwickelt, welches erlaubt Haare zu modellieren und animieren.

Es wurde nun versucht, dieses Programm auf Gräser sprich Wiesen anzuwenden. Es hat ansprechbare Lösungen ergeben. Das einzige Problem liegt bei dem grossen Speicherplatzverbrauch.
Eine Fläche von 3.1 x 3.1m mit produzierten Grashalmen (10cm Abstand / Abschnitte=4 / Länge=20cm / Seitenflächen=6) ergeben 2MB.

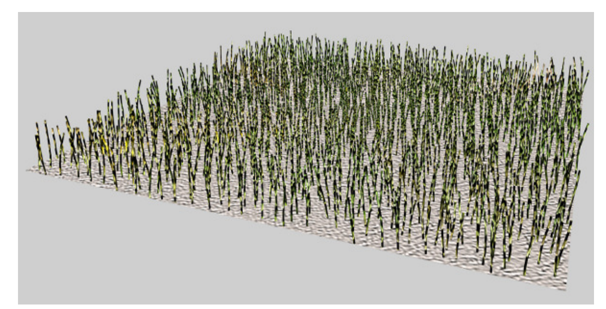
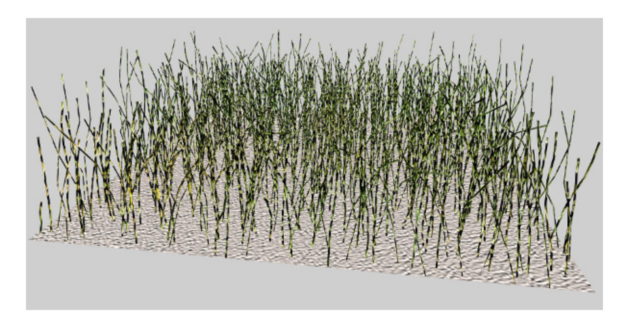



Abbildung 39 Beispiele von Hair Departement

Ein lustiges Tool zum Spielen, vor allem weil zur Animation verschiedene Modifikatoren zur Verfügung gestellt werden.
Für realitätsnahe Wiesen ist das Plug-In nicht geeignet. Mann müsste derart viele Haarskelette erzeugen, dass es auf einer grösseren Fläche schlicht den Speicherrahmen eines PC sprengen würde.

5.3 Schlussfolgerungen

Es hat sich gezeigt, dass einfach Möglichkeiten bestehen realistische Bäume in eine 3D-Szene zu integrieren. Der Aufwand kann allerdings stark variieren und zum Teil übers Ziel hinaus schiessen. Es macht keinen Sinn einen Wald akribisch aus Laserscannigdaten aufzubauen wenn die fertige Szene nur zur Visualisierung genützt wird und nicht zur Untersuchung von ökologischen Gründen (Lebensraumuntersuchung, Studien von Wachstumsreaktionen auf Wind und andere Umwelteinflüsse, etc.) oder ökonomische Gründe (Bauholzvolumenschätzung für die Holzproduktion, Ermittlung und Quantifizierung von Verlusten während des Baumwachstums, etc.)[21]/[22].
Für eine realitätsnahe Visualisierung eines Waldes müssen nicht alle bestehende Bäume modelliert werden, sonder nur darauf geachtet werden, dass der Wald als geschlossen interpretiert wird. So kann die Datenmenge und nicht zuletzt die Performance sowie die Datenverwaltung in einem angemessenen Rahmen gehalten werden. Am Waldrand ist es erstrebenswert eine exakte Abbilddung der Realität wiederzugeben. Im Speziellen bei allein stehenden Bäumen ist es sogar ein Muss. Dort darf auch ein genaues Augenmass auf die Form gegeben werden. Hier besteht die grösste Schwierigkeit, denn ein allein stehender Baum kann mit einer Vegetationsbibliothek nie so genau verformt werden, dass er die gleiche Erscheinung aufweisst, wie in Natura. Die Rekonstruktion aus Laserscanningdaten könnte hier weiterhelfen, im Bewusstsein, dass zwar bei der Visualisierung Abstriche gemacht werden müssen, dafür aber die Form genau stimmt.
Als noch schwieriger stellt sich die Modellierung von einer Wiese dar. Ein Orthophoto ist zwar wie bei einem vergebenen Material im Nahbereich schwach, doch eine Orhophoto-Texturierung wird als besser betrachtet. Das unter Kapitel 5.2.4 erwähnte Plug-In, welches laut Hersteller wie die Vegetationsbibliothek ebenfalls eine überraschende Detailschärfe hat, wäre eine lohneswerte Alternative.

6 Videothematik

Verfasser: D. Emmenegger

Die bis jetzt erstellten Szenen sehen als Standbilder gut aus, aber eine Animierung in Echtzeit ist heute noch nicht möglich. Deshalb werden die Bilder vorgerendert und anschliessend zu einem Film zusammengesetzt. Dieses Kapitel soll eine Einführung in diese Thematik bieten.

6.1 Theoretische Grundlagen

Im Zusammenhang mit Film und Video tauchen immer wieder verschiedene Begriffe auf. Nachfolgend werden die häufigsten in Cinema 4D verwendeten Begriffe kurz erklärt werden.

6.1.1 AVI File

Das AVI-Format ist ein sogenanntes "Containerformat". Dies bedeutet, dass man ein File hat, wo man seine Daten hinein gibt. Die jeweilige Codierung der Daten spielt dabei keine Rolle. Auch können die Filmdaten anders codiert sein als die Audiodaten.

6.1.2 Codec

Als Codec's bezeichnet man Verfahren, mit denen Daten digital **co**diert und **dec**odiert werden können. Meist werden die Daten dabei nicht verlustfrei codiert. Der ganze Codierungsprozess benötigt viel Rechenzeit, da zum einen die Daten reduziert werden und gleichzeitig eine Datenkompression stattfindet.

Die gebräuchlichsten Codec's sind:

- XviD (Video)
- Lame (Audio)

6.1.3 PAL (Bildformat)

Unter PAL wird neben vielen anderen Anwendungen auch ein Bildformat verstanden. Dabei handelt es sich um den in Europa weit verbreiteten Fernseh-Standart. Die Pixel der einzelnen Bilder sind dabei quadratisch und haben ein Seitenverhältnis über das ganze Bild von 4:3. Bezieht man dies nun auf ein ganzes Bild, so erhält man eine Auflösung von 768x576 Pixel. Für die Erzeugung von Filmen benützt man am besten dieses Format, da es mit den meisten Bildschirmformaten übereinstimmt. Es können keine Bildverzerrungen mehr entstehen.

6.2 Untersuchungen und Resultate

Die technischen Aspekte, von einer Videoanimation sind heute mehrheitlich gelöst. Deshalb zielten unsere Untersuchungen in die Richtung, wie eine Videoanimation gemacht werden kann. Schnell kommt man dabei in die Thematik des Films, was sehr interessant, aber auch sehr umfangreich ist. Deshalb konnte in den Untersuchungen eigentlich nur an der Oberfläche "gekratzt" werden.

6.2.1 Empfehlungen zu Videoanimationen

Aus Gesprächen mit Personen von der HGK Basel, welche zum Projekt " Architektur und Film" gehören, ergaben sich einige sehr interessante Empfehlungen, wie eine Video-animation zu gestalten ist.

6.2.1.1 Idee und Storyboard

Grundsätzlich soll man sich vor dem "Filmen" in Cinema 4D einige Gedanken über den Film machen. Dazu gehören z. B. :

- Welches ist mein Zielpublikum?
- Wie lange soll der Film sein?
- Was will ich zeigen?
- Wo will ich den Film vorführen?

Dies sind ganz entscheidende Aspekte, welche man zuerst beantworten muss. Daraus ergeben sich Konsequenzen für die Kameraführung, den Schnitt und die Vertonung.

Wenn diese Grundfragen geklärt sind, kann man sich an den Ablauf des Filmes wagen. Hierzu macht man sich am besten im Team Gedanken über die Führung der Kameras und die Positionen der Schnitte. Zeichnungen in Form eines Storyboards sind hierzu sehr nützlich. Es braucht zu Beginn etwas Übung, die Zeichnungen sauber anzufertigen. Das Verständnis für den Film wird dadurch aber wesentlich gesteigert.

Personen und Objekte, bei welchen die Grössenverhältnisse bekannt sind, fördern die Massstäblichkeit. Es wird ein besserer Eindruck der Grössenverhältnisse geschaffen. Deshalb sollten sie, wenn möglich, in eine Szene integriert werden. Selbst wenn diese nur statisch vorhanden sind reicht es für den Grössenbezug aus.

Heute werden immer bessere und perfektere Szenen geschaffen. Nur wirken diese immer künstlicher. Was ihnen fehlt, sind die natürlichen kleinen Unperfektheiten:"z.B. das Wackeln der Kamera oder das Flimmern des Filmes". Auch ein nicht ganz perfekter Schwenk ist meist besser und natürlicher.

6.2.1.2 Kamerapfad und Sichten

Kamerapfade und Sichten sind natürlich zu wählen. Ein unnatürliches Bewegen wie z.B.

- Seitwärtsbewegungen
- bei Flügen starker Seitenblick

wirkt komisch und soll vermieden werden. Das Objekt soll aus Sichten und Positionen gezeigt werden, welche in der Realität auch von einem Kameramann eingenommen werden könnten. Auch gehört ein Beschleunigen und Abbremsen bei Kameraschwenkungen dazu. So sieht die Szene gleich viel realitätsgetreuer aus.
Auch viel Zoomen ist unter Filmemachern verpönt. Zudem bringt es nicht den gleichen Effekt wie eine reale Kamerafahrt. Am besten entschliesst man sich am Anfang für eine Brennweite und "filmt" damit die ganze Szene.

6.2.1.3 Schnitte

Schnitte zwischen einzelnen Szenen sollen einfach sein. Spielereien wie es heutige moderne Videobearbeitungsprogramme ermöglichen, sind unangebracht und wirken verspielt. Meist genügt eine einfache Überblendung oder ein Ausgang zu Schwarz.

6.3 Schlussfolgerungen

Die technischen Grundlagen und Programme sind in diesem Gebiet vorhanden. Die Erstellung und Verarbeitung von grossen Datenmengen stellen auch kein Problem dar. Die Hindernisse zu einer guten Videoanimation sind nicht im technischen Bereich zu suchen. Vielmehr liegen sie in der Fantasie und Kreativität des Erstellers. Bei dieser Thematik treffen zwei gegensätzliche Gruppen aufeinander:

- Die Geomatiker, die ihre Stärken in der detailgetreuen Modellierung von Objekten sehen.
- Filmemachern und Künstlern, welche ihre Stärken eher in der Kreativität haben.

Eine Zusammenarbeit zwischen beiden Gruppen kann zu kleineren Missverständnissen führen. Eine Zusammenarbeit sollte aber unbedingt gefördert werden.

Trotz den getätigten Untersuchungen und Umsetzungen kann der jetzige Stand nicht endgültig sein. Es hat sich gezeigt, dass die Thematik weit grösser ist als zu Beginn angenommen. Deshalb sind weitere Forschungen in diesem Gebiet unbedingt notwendig. Die erste Zusammenarbeit mit der Projektgruppe "Architektur und Film" der HGK Basel bestätigte dies.

Betrachtet man die Zielsetzung zu diesem Themenbereich, so wurden keine technischen Lösungen gemacht. Vielmehr wurde eine Vorgehensweise aufgezeigt, wie diese Thematik zu erfüllen ist.

7 Visualisierung im Internet

Verfasser: Th. Schuler

7.1 Übernahme von 3D-Modellen in die Software G-Vista Suite 2005

G-Vista Suite

Es wird eine 3D-Szene über einen Höhenraster sowie ein Orthophoto erstellt. Darin wird das texturierte Objekt als VRML1-File importiert. Für diesen Arbeitsschritt wird der G-Vista Builder eingesetzt. Er wird für die gesamte Erstellung der 3D-Szene mit allen zusätzlichen Inhalten verwendet. Für die Visualisierung der 3D-Lanschaften in Echtzeit via CD/DVD oder Internet wird der G-Vista Viewer benützt.

Untersuchungen

G-Vista Builder erstellt die Szene nicht an den Landeskoordinaten.
Das eingelesene VRML1-File Objekt muss über eine MatrixPos-Node an die richtige Position verschoben werden.
Dabei müssen im *.gsl-File unter <TransformationMarix> die Werte der Koordinatenverschiebung durch den Skalierfaktor dividiert werden. Die ausgerechneten Werte werden nun im Scene-Manager im MatrixPos-Node angegeben.
Die Texturen werden nicht ins VRML1-File integriert, sondern extern gespeichert und darauf verwiesen. Die einzelnen Texturbilder weisen eine Grösse von 1024 x 681 Pixel auf. Sie werden nicht verkleinert, da ihre Ausdehnung zum Teil über eine ganze Fassade reichen. Bei einer verkleinerten Pixelauflösung würde dies bei naher Distanz bemerkt werden. Die entstandene Szene ist als xml File (gsl) zwar um die 20Megabytes gross, beim abspeichern als binary File (gsx) nur noch ca. 650Kilobytes.

Import über 3StudioMax in den G-Vista Builder

Die Firma GEONOVA AG hat ein „Geonova Max Export Plug-In“. Da das Importieren des Schlosses Wildenstein über ein VRML-File direkt ohne Probleme erfolgt, wir dieses Plug-In nur zur Verwendung eines Exportes von der Vegetation miteinbezogen.
Es hat sich herauskristallisiert, dass eine solche Szene am besten über ein VRML2-File von Cinema 4D exportiert wird und in 3DStudioMax importiert wird. So sind zwar keine Geometrieverluste zu beklagen, leider geht aber die Texturierung verloren. Es wird nur ein Material mitexportiert, welches sich nun auf den gesamten Bereich der Baumpolygone ausdehnt. Die Bäume sehen zwar immer noch besser aus als jede unter Kapitel 5.1.1 erwähnte Darstellungsform, doch es kann geradeso gut darauf verzichtet werden und eine einheitliche Farbe vergeben werden. Beim erneuten Export über das erwähnte Plug-In wird ein binary File (gsx) erstellt, welches anschliessend direkt in der Szene eingefügt wird. Es kann kein xml File (gsl) exportiert werden. Erstens ist die entstehende Datenmenge viel zu gross und anderseits braucht es zu viel Zeit beim exportieren, was oft zu Abstürzen des Programms führt. Den direkten Weg über ein VRML1-File funktioniert zwar für einzelne Bäume, doch ein VRML-File ist schnell sehr gross und dementsprechende blast es auch die G-Vista-Szene unnötig auf.

7.2 Möglichkeiten zur Vereinfachung von Geometrie und Texturen

Geometrie

Wie in Kap. 8 beschrieben wird, kann die Dreiecksvermaschung bereits in Cyclone und später in Cinema 4D (siehe Kap. 2.7) ausgedünnt werden. Die Geometrie darf bei diesem Prozess aber auf keinen Fall verfälscht werden. Es ist ein guter Mittelweg zu finden.

Texturen

Wird in Kap. 4.2.4 ausführlich behandelt.

7.3 Schlussfolgerungen

Die Erstellung einer exakt gleichen Szene wie in Cinema 4D kann teilweise erfüllt werden. Die Übernahme von Texturen stellt kein Problem dar. Die bereits vergebenen und zugewiesenen Materialien und Mappings der Bäume in Cinema 4D gehen beim Export aus 3DstudioMax verloren. Das Material dehnt sich nun über die gesamten Polygone hinweg.
Der Export über 3DStudioMax hat den einzigen Vorteil, dass ein binary File (gsx) erstellt werden kann, was das Überführen der Daten in G-Vista Builder erleichtert. Anderseits benötigt dieser Weg enorm viel Zeit.

8 Funktionalitätsvergleich der verwendeten Programme

Verfasser: Th. Schuler

Programm	Funktion / Arbeitsschritt	Vorteil	Nachteil
Cyclone	Aufteilen der Punktwolke	• Navigieren in der Punktwolke • Identifizierung der einzelenen Teilobjekte • Mehrere Personen können am gleichen Objekt arbeiten	• viele Model Spaces (Views)
Cyclone	Konstruktion von Linien	• eine Linie kann in mehren Punktwolken verwendet werden; Dreiecksvermaschung von verschiedenen Teilobjekten stimmen an den Rändern überrein. Keine mühsame Überarbeitung beim späteren Zusammenfügen	• Es können nur erfasste Punkte für eine Linie verwendet werden • Wenn ein Punkt verschoben wird, muss die Lage im Raum überprüft werden • Man hat nicht so viele Konstruktionsmöglichkeite n wie in einem 3D-CAD Programm
Cyclone	Dreiecksvermaschung über Punktobjekt	• Automatisch; einfach	
Cyclone	Äusdünnung des Maschenmodells	• Veränderung der Geometrie kann graphisch Überprüft werden	• Es müssen an den Ränder der Punktwolke Linien definiert worden sein, sonst gibt es Verfälschungen in der Form • Änderungen können nur durch die Undo-Taste vorgenommen werden
Micro Station	Zusammenfügen der Teilobjekte	• „Standard-CAD" mit vielen Tools • Layerstruktur	• komplex
Micro Station	Konstruktion / Details	• Grundwissen vorhanden	• 3D-Konstruktion nicht immer einfach
Micro Station	Erstellen eines Filmes (Kamerapfad)	•	• schwierig
iWitness	Orientierung der Bilder	• Einfach, schnell und unproblematisch	• Lösung muss nicht korrekt sein. Kann allerdings graphisch überprüft werden.
iWitness	Konstruieren / Messen	• angenehm und benutzerfreundlich	• Es können nur Linien auf gemessene Punkte konstruiert werden
PHIDIAS-MS	Orientierung der Bilder	• wenn Ori i.o.; gute, sichere Lösung	• aufwendiger, zeitintensiver als IWitness

PHIDIAS-MS	Konstruieren / Messen	•siehe Micro Station	
Cinema 4D	Ausdünnung des Maschenmodlles	•einfach •kann Teilobjektweise unterschiedlich stark gewählt werden	
Cinema 4D	Erstellen eines Filmes (Kamerapfad)	•einfach	
Cinema 4D	Texturierung	•es müssen nicht einzelne Polygone texturiert werden, sonder es können ganze Fassaden texturiert werden	•Orientierungswerte der Aufnahmen sind nötig da bei der Texturierung Kameras definiert werden müssen
DILAS	Texturierung		•Jedes einzelne Polygon muss texturiert werden •Polygone müssen streng im Gegenuhrzeigersinn erfasst worden sein

Tabelle 5 Funktionsvergleich

Überschneidungen

- Orientierung / Auswertung in IWitness und PHIDIAS-MS → kann nicht vermieden werden. Kommt auf die Aufgabenstellung, die Sorgfalt bei der Planung und Durchführung der einzelnen Projekte drauf an.
- Ausdünnung des Maschenmodells in Cyclone, Internet Model Optimizer und Cinema 4D → Eine Dreiecksreduktion in Cyclone muss vorgenommen werden, da sonst über alle aufgenommenen Laserscanningpunkte Dreiecke entstehen. Endgültige Reduktion in Cinema 4D
- Erstellen eines Filmes; anlegen eines Kamerapfades in MicroStation als auch in Cinema 4D → aus eigener Erfahrung aus dieser Diplomarbeit und von früher nur noch in Cinema 4D vornehmen.

Möglicher Ablauf um Überschneidungen zu vermeiden oder zu reduzieren

- Dreiecksvermaschung bei Laserscanningdaten in Cyclone mit einer ersten Ausdünnung des Maschenmodells
- Orhophotogenerierung in LPS
- DGM-Generierung in InRoadSite
- Orientierung in IWitness
- Übernahme der Orientierung in PHIDIAS-MS (FHBB arbeitet an einer Lösung)
- Konstruktion in PHIDIAS-MS
- Zusammenfügen der einzelnen Teilen, Konstruktion von Details und sinnvoller Layerstruktur in MicroStation; KONSTRUKTIONSSOFTWARE
- Texturierung, Visualisierung und Animation in Cinema 4D

Begründung der Auswahl einzelner Programme für zugewiesene Arbeitschritte

Cyclone:	bedienerfreundliches Programm Die wenigen verschiedenen Schritte / Funktionen / Schaltflächen sind schnell erlernt
IWitness	bedienerfreundliches fast selbsterklärendes Programm
PHIDIAS	Oberfläche von MicroStation bekannt
MicroStation	In der FHBB Bestandteil des Unterrichtes

9 Umsetzungen und Produkte

Verfasser: Th. Schuler und D. Emmenegger

Dieser Bericht beinhaltete bis jetzt nur spezifische Untersuchungen und Tests, welche in dieser Diplomarbeit gemacht wurden. Dabei wurde in den Hintergrund gestellt, dass eine Aufgabe auch darin bestand, die Untersuchungen an konkreten Daten praktisch umzusetzen. Dies war zum einen das Schloss Wildenstein und zum anderen die Ruine Homburg. Beide Projekte wurden jeweils in Blockkursen des 6. Semesters der Abteilung Vermessung und Geoinformatik der FHBB erarbeitet. Die zur Verfügung gestellten Grundlagen der beiden Projekte waren beinahe identisch. Es waren dies insbesondere:

- Maschenmodell der Objekte
- Höhenmodell als 25m Raster mit Verdichtungen in der näheren Umgebung der Objekte
- Nahbereichphotogrammetrieaufnahmen (in PHIDIAS Orientiert)
- Orthophotos mit einer Auflösung von 0.5m
- digitale Fotos



Abbildung 40 Standbild von Schloss Wildenstein

Ein erwähnenswerter Unterschied ist, dass am Schloss Wildenstein mehrere Jahrgänge daran gearbeitet haben. Die Ruine Homburg ist dagegen erst von der Klasse G2002 erfasst worden. Darum waren die Grundlagen beim Schloss Wildenstein umfangreicher. Das Schloss Wildenstein stellt grundsätzlich einen interessantern Datensatz dar, da die Komplexität des Objektes grösser ist. Es sind viele verschiedene Gebäude, Fassaden, Dächer, Kamine, Nischen und eine Gartenanlage modelliert. Dies war der Grund, warum Wildenstein uns als Pilotprojekt diente.

Infolge des Pilotprojektes wurden vom Schloss Wildenstein die folgenden Produkte erzeugt:

- Cinema 4D Modelle im Baukasten Prinzip
  - Schloss mit Texturen
  - DGM mit Orthophoto
  - Vegetation (Bäume)
- 2.5 Min. Videoanimation
- G-Vista Szene
- div Standbilder des Schlosses

Da die Grundlagen bei der Ruine Homburg nicht ganz so ausgereift sind, waren einige Korrekturen der Grundlagedaten nötig welche uns Zeit raubten. Weiter ist anzufügen, dass die Ruine nur aus 4 Fassaden besteht mit zusätzlich vielen kleinen Mauern. Auf der anderen Seite lagen uns vom gesamten Objekt mit Umgebung terrestrische Laserscanningdaten zur Verfügung. Diese Umstände führten dazu, dass nicht alle gewünschten Produkte in voller Qualität erzeugt werden konnten. Trotzdem wurden die folgenden Produkte erstellt:

- Cinema 4D Modelle im Baukasten Prinzip
  - Ruine mit Aussentexturierung
  - DGM
  - Vegetation (Bäume); Wald westlich der Burg vollständig erfasst. Im Süden und Osten mit Löchern
- 1.5 Min. Videoanimation
- div Standbilder der Ruine

Da bei den Aufnahmen der Ruine Homburg ein Laserscanner eingesetzt wurde, konnten hier die Bäume exakt so rekonstruiert werden, wie sie in Natura sind. Dies war beim Schloss Wildenstein nicht möglich. Hingegen waren genau diese Bäume hinderlich für eine gute Aufnahme der Fassaden mit Nahbereichsphotogrammetrie. Darum konnten nicht alle Teile der Ruine texturiert werden.

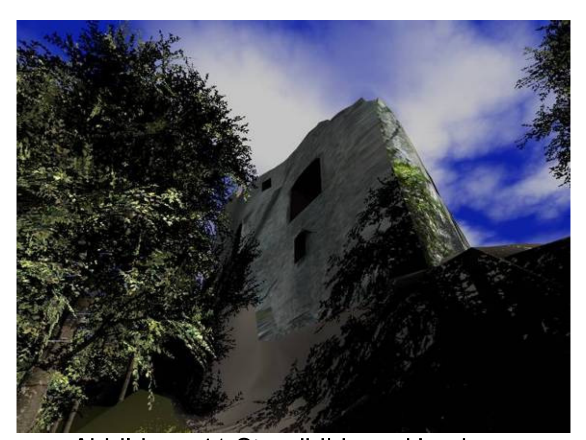


Abbildung 41 Standbild von Homburg

Ein Punkt, welcher bei beiden Projekten negativ aufgefallen war, ist, dass nie explizit Fotoaufnahmen zum Texturieren gemacht wurden. Es waren jeweils nur Aufnahmen vorhanden, welche zum Auswerten geeignet sind. Dadurch sind die erzielten Resultate bei der Texturierung noch steigerungsfähig.

Da bei beiden Projekten keine Aufnahmen über den Vegetationsbestand gemacht wurden, mussten diese Informationen mühsam zusammengesucht werden. Dazu dienten die Fotos des Photogrammetrieprojektes, die Laserscanningdaten oder auch das vorhandene Orthophoto. In Zukunft währe ein einfacher Feldhandriss mit ein paar Vermassungen (Abstand, Höhe, Baumkronendurchmesser, Art ...) hilfreich.

Abschliessend kann gesagt werden, dass mit den vorliegenden Daten gearbeitet werden konnte. Die grundlegenden Abläufe und Vorgehensweisen bei der Visualisierung von komplexen, kulturhistorischen Bauten konnten getestet und optimiert werden. Wenn es auch zum Teil erhebliche Schwierigkeiten gab. Diese lagen vor allem in der Art wie diese Daten abgelegt und aufgebaut waren. Ein paar Stunden am Ende eines Projektes in die Archivierung und Strukturierung der Daten zu investieren, wären darum eine zwingend lohnende Investition, welche bereits im Projektmanagement integriert und vom Projektverantwortlichen überprüft werden muss.

10 Schlusswort

Verfasser: Th. Schuler

In diesem Abschnitt soll ein Überblick und eine Zusammenstellung der Resultate und Erkenntnisse über die einzelnen Kapitel gegeben und in einen Gesamteindruck zusammengefasst werden. Die folgende Aufzählung der einzelnen Teilaspekte soll zunächst nur die wichtigsten Kernaussagen auflisten.

Textur

- Eine Texturierung von komplexen 3D-Geoobjekte ist in machbarer Zeit möglich.
- Ein Export ohne Texturverlust in das VRML-Format ist ebenfalls möglich.

Vegetation

- Es können Bäume aus einer Vegetationsbibliothek an exakter Position gesetzt und in ihrer Form angepasst werden.
- Die automatische realitätsnahe Modellierung aus Laserscannerdaten ist noch nicht praktisch umsetzbar.
- Die Darstellung einer Wiese im Nahbereich ist bis heute nicht gelöst.

Video

- Die „Cinema 4D-Szenen" können mittels Filmdarstellung präsentiert werden.
- Bei der Realisierung fand eine Zusammenarbeit mit dem Forschungsteam der HGK (http://www.fhbb.ch/hgk/af/architektur_und_film/) statt. Die in kürzester Zeit erlernten gestalterischen Möglichkeiten ergaben eine Horizonterweiterung die sich in den erstellten Filmen positiv niederschlugen.

Datenfluss

- Ein optimaler, effizienter Ablauf von der Planung bis zur Animation für die Bereitstellung von kulturellen wertvollen Objekten wurde erstellt.
- Überschneidungen wurden aufgedeckt und die Wirtschaftlichkeit wird durch den erstellten Datenfluss gesteigert.

Visualisierung in G-VISTA SUITE 2005

- Die Übernahme der 3D-Objekte aus Cinema 4D in die Echtzeit 3D-Geovisualisierungs-software G-Vista der Firma Geonova, ist mit VRML1 möglich.

Ausblick/Gesamteindruck

Die erzielten Resultate und die Verwirklichung an den konkreten Datensätzen zeigen einen weiteren Schritt in eine Richtung der realitätsnahen Abbildungen. Insbesondere die nahezu unbegrenzten Möglichkeiten von Cinema 4D im Bereich der Animation konnte wegen der beschränken Zeit nicht voll ausgekostet werden.
Wir haben einen Grundstein gesetzt, der unbedingt erweitert werden muss.

Diese Diplomarbeit war für uns das erste von Anfang an selbständig bearbeitende grosse Projekt an der FHBB. Es zeigte uns, dass ein gutes und eingehaltenes Projektmanagement überlebenswichtig ist. Da die Aufgabenstellung relativ klar formuliert war, konnte eine klare Aufgabenteilung vorgenommen werden, was sehr hilfreich war.

11 Quellenverzeichnis

11.1 Literaturverzeichnis

[1] J. Encarnação, W. Strasser, R. Klein, *Graphische Datenverarbeitungen 1*, 4. Auflage, Oldenbourg 1996

[2] J. Encarnação, W. Strasser, R. Klein, *Graphische Datenverarbeitungen 2*, 4. Auflage, Oldenbourg 1997

[3] Rüdiger Mach, *3D Visualisierung*, 1. Auflage, Galileo 2000

[4] Helmut Hofmüller, Martin Seiwert, *Digital Audio/Video*, Addision Wesley Longmann 1999

[5] Thomas Luhmann, *Nahbereichsphotogrammetrie*, 2. Auflage, Wichmann 2003

[6] Maxon Computer GmbH, *Cinema 4D Referenzhandbuch R8*, Maxon 1989-2001

[7] Maxon Computer GmbH, *Cinema 4D Tutorial-Handbuch, Net Render, Mocca, Pyro Cluster, Dynamics, Advanced Render, Thinking Particles*, Maxon 1989-2002

[8] Maxon Computer GmbH, *Cinema 4D Release 9 Quickstart Handbuch* , Maxon 1989-2004

[9] Bentley Systems, *Microstation V8 Benutzerhandbuch*, Bentley 2000

[10] Bentley Systems, *Microstation V8 Einführung für Benutzer von Microstation*, Bentley 2000

[11] L. Grammatikopoulos, I. Kalisperakis, G. Karras, T. Kokkinos, E. Petsa, *Automatic multi-Image photo-texturing of 3D surface models obtained with laser scanning,* CIPA international workshop Lisbo 25-27 October 2004

[12] S. El-Hakim, L. Gonzo, M. Picard, S. Girardi, A. Simoni, E. Paquet, H. Viktor, C. Brenner, *Visualisation of highly textured surfaces,* Archaelogy and intelligent cultural heritage 2003

[13] S. El-Hakim, E. Whiting, L. Gonzo, S. Girard, *3D-reconstruction of complex architectures from multiple data*, 22-24. August 2004

[14] S. El-Hakim, E. Whiting, L. Gonzo, *3D modelling with reusable and integrated building blocks, Conference on Optical 3-D Measurement Techniques*, Austria 2005

[15] J-A Beraldin, M. Picard, S.F. El-Hakim, G. Godin, V. Valzano, A. Bandiera, C. Latouche, *Virtualizing a Byzantine Crypt by Combining High-resolution Textures with Laser Scanner 3D Data*, VSMM Korea 2002.

[16] Andreas Muhar, *Dreidimensionale Visualisierung von Vegetationsbeständen auf unterschiedlichen Massstabsebenen,* F. Dollinger und J.Strobl (1996), Selbstverlag des Instituts für Geographie der Universität Salzburg.

[17] Olaf Büscher, *Die 3D-Visualisierung einer Lippeauen-Renaturierung mit dem World Construction Set (WCS),* Universität München 2004

[18] Oliver Deussen, Bernd Lintermann, *Interactive Modelling of Plants*, Universität Magdeburg Januar/Februar 1999

[19] Oliver Deussen, *Nichtrealistische Computergraphik*, Informatik Spektrum 24. April 2001, Springer-Verlag Berlin Heidelberg

[20] Oliver Deussen,k Pat Hanrahan, Bernd Lintermann, Radomir Mech, Matt Pharr, Przemyslaw Rusinkiewicz, *Realistic modelling and rendering of plant ecosystems*, To Appear in Proceedings of SIGGRAPH 98

[21] Norbert Pfeifer, Ben Gorte, Daniel Winterhalder, *automatic reconstruction of single trees from terrestrial laser scanner data*,

[22] Ben Gorte and Norbert Pfeifer, *structuring laser-scanned trees using 3D mathematical morphology*, ISPRS-XXth Congress – Commission 5, Working Group 2

[23] George Vosselman, *3D reconstruction of roads and trees for city modelling*, ISPRS-XXth Congress – Commission3, Working Group 3/3

[24] Bettina Geiger, Karin Egger, Andreas Muhar, *Integrierte 3D-Visualisierungs-Systeme für die Landschaftsplanung: Konzepte und Marktrealität*, Institut für Freiraumgestaltung und Landschaftspflege, Universität für Bodenkultur Wien

11.2 Softwareverzeichnis

Bentley Systems (2002) MicroStation, Version 08.05.00.64, USA: Bentley Systems

Maxon Computer GmbH, Cinema 4D Studio Bundle, Version 9.507, Deutschland 2005

Adobe, Adobe Premiere Pro, Version 1.5. USA 2004

Microsoft, Microsoft Office 2003, USA 2003

Autodesk, 3DS Max , Version 8, USA 2005

Geonova AG, G-Vista Builder 2005 Professional, Version 1.0.0.136, Muttenz 2005

Geonova AG, G-Vista Viewer 2005, Version 3.2.0.1, Muttenz 2005

Jasc Software, Paint Shop Pro, Version 6.00, USA 1999

IDM Computer Solutions, Ultra edit-32 Professional, Version 7.20a, USA 2000

Leica Geosystems, Cyclone, Version 5.3 (build 630), Schweiz 2005

Phocad Ingenieurgesellschaft, Phidias-Ms, Version 2.5.3, Deutschland 2005-11-15

Open Source, PTlens, http://epaperpress.com/ptlens/index.html (14. Oktober 2005)

Leica Geosystems, Erdas Imagine 8.7 Service Pack 2, August 2005

11.3 Datenverzeichnis

Aus den folgenden Projekten wurden uns Daten zur Verfügung gestellt:

Projekt Ruine Homburg, FHBB Abteilung Vermessung und Geoinformation 2005

Projekt Schloss Wildenstein, FHBB Abteilung Vermessung und Geoinformation 2005

Projekt Schloss Ebenrain, Hochschule für Technik und Wirtschaft Dresden 2001

Projekt Schloss Zwingen, FHBB Abteilung Vermessung und Geoinformation 2001

11.4 Abbildungsverzeichnis

11.5 Formelnverzeichnis

11.6 Tabellenverzeichnis

12 Anhang

12.1 Mastertabelle

	Y[m]	X[m]	Z[m]		Y[mm]	X[mm]
Shift	622300	253200	0	**Hauptpunkt**	-0.2394	0.115
	Y[mm]	**X**[mm]				
Bildgrösse	15	22.6		**Brennweite**	27.1744	mm

Bild Nummer	DSC_0037										
Y	**X**	**Z**	**Omega**	**Phi**	**Kappa**						
622368.816	253259.663	529.727	76.0068	339.3773	382.3136			(o,p,k) in rad	1.19	5.33	6.01
								Werte der standart Rot. Matrix	-0.5	-0.2	0.03
Y	**X**	**Z**	**Heading**	**Pitch**	**Bank**	**Offset Y**	**Offset X**	(h,p,b) in rad	0.89	-0.2	0.03
68.816	59.663	529.727	56.674	-12.322	1.947	-1.596	0.509				

Bild Nummer	DSC_0034										
Y	**X**	**Z**	**Omega**	**Phi**	**Kappa**						
622445.231	253230.665	530.924	82.0888	9.4405	4.1112			(o,p,k) in rad	1.29	0.15	0.06
								Werte der standart Rot. Matrix	-1	-0.3	0.02
Y	**X**	**Z**	**Heading**	**Pitch**	**Bank**	**Offset Y**	**Offset X**	(h,p,b) in rad	0.18	0.25	0.02
145.231	30.665	530.924	11.271	15.938	1.246	-1.596	0.509				

Bild Nummer	DSC_0032										
Y	**X**	**Z**	**Omega**	**Phi**	**Kappa**						
622489.041	253250.434	531.8	73.8513	49.4585	20.3011			(o,p,k) in rad	1.16	0.78	0.32
								Werte der standart Rot. Matrix	-0.7	-0.3	0.02
Y	**X**	**Z**	**Heading**	**Pitch**	**Bank**	**Offset Y**	**Offset X**	(h,p,b) in rad	0.75	0.26	0.02
189.041	50.434	531.800	47.430	16.543	1.282	-1.596	0.509				

Bild Nummer	DSC_0033										
Y	**X**	**Z**	**Omega**	**Phi**	**Kappa**						
622468.715	253239.802	531.059	78.4052	30.0065	7.5055			(o,p,k) in rad	1.23	0.47	0.12
								Werte der standart Rot. Matrix	-0.8	-0.3	-0
Y	**X**	**Z**	**Heading**	**Pitch**	**Bank**	**Offset Y**	**Offset X**	(h,p,b) in rad	0.46	0.27	-0
168.715	39.802	531.059	29.309	17.245	-2.327	-1.596	0.509				

12.2 Typische Programm Abläufe

Es werden Grundlagen Kenntnisse bei den verwendeten Programmen vorausgesetzt.

12.2.1 Texturierung

Verfasser: D. Emmenegger

Es wird der Ablauf beschrieben, wie man mit Cinema 4D Fotos auf ein Polygon Objekt "kleben" kann.

Grundlagen

Um eine Texturierung erfolgreich durchführen zu können werden folgende Grundlagen benötigt:

- Maschenmodell des zu texturierenden Objektes, aufgeteilt in die einzelnen Teilobjekte.
- ausgefüllte Mastertabelle
- verzeichnungskorrigierte Fotos

Ablauf

Zuerst werden die Kameras definiert, welche für das Kameramapping verwendet werden.

1. **Befehls-Gruppe**
   - Objekte
     - Szene-Objekte
       - Kamera

Danach wird der Kamera im Objektmanager die Nummer des zu texturierenden Bildes vergeben. Anschliessend werden die Koordinaten und Raumwinkel für die jeweilige Kamera im Attribut-Manager eingetippt. Dabei können gleich noch die Objektiv-Parameter eingegeben werden. Dieser Vorgang ist für alle Kameras zu wiederholen.

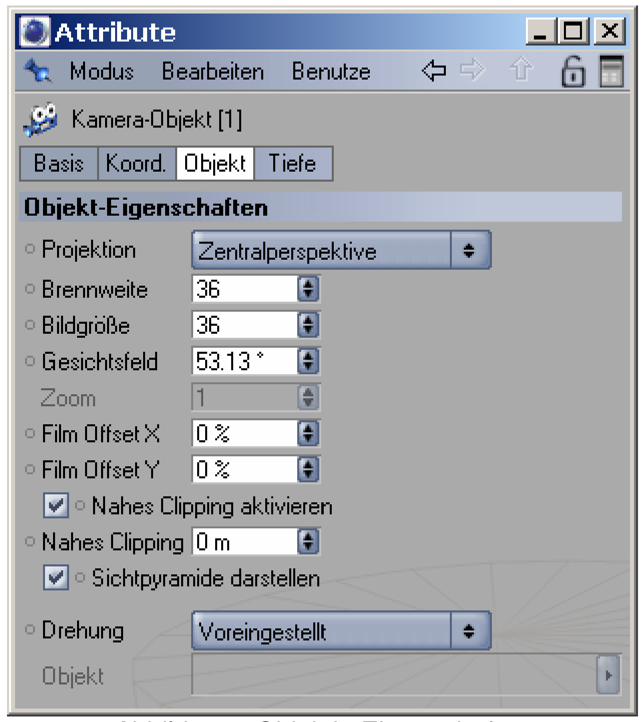


Abbildung : Objektiv-Eigenschaften

Es sind weiter folgenden Eingaben im Attribute-Manager zu machen:

Projektion: Zentralperspektive

Brennweite: die Brennweite aus der Kamerakalibrierung

Bildgrösse: die Grösse des CCD-Chips [mm]

Wenn man die Szene in der 3D-Ansicht anschaut, sollten nun die definierten Kameras zu sehen sein.
Um eine Texturierung vornehmen zu können, müssen die einzelnen Fotos noch als Material definiert werden. Dies findet im Material-Manager statt.

2. Material-Manager

Datei

Neues Material

Im Attribut-Manager kann nun das Foto als Textur zugewiesen werden.



Abbildung : Material im Attribut-Manager

Zu überprüfen ist noch, dass unter der Registerkarte "Basis" die Option "Glanzlicht" deaktiviert ist. Ansonst werden die Fotos mit einem unnatürlichen Glanzlicht dargestellt. Um die Übersicht zu erhalten, sollte das Material gleich benannt werden wie die Kamera. Dies kann auch unter der Registerkarte "Basis" gemacht werden.

Anschliessend kann durch einfaches Drag&Drop das Material dem Polygonobjekt im Objekt-Manager zugewiesen werden.

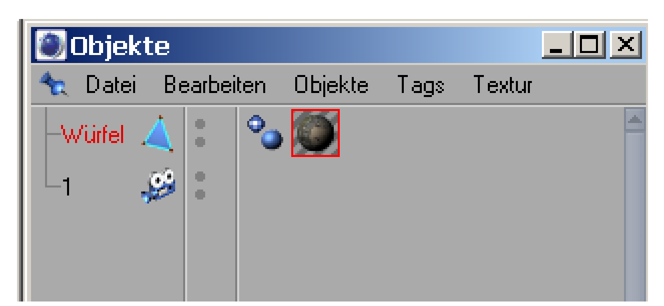


Abbildung : Objekt- Manager mit Objekt, Material und Kamera

Bei einem Doppelklick auf das Material im Objektmanager kommt man zu den "Textur-Tag" Eigenschaften des Materials. Dort kann man die Projektionseigenschaften des Materials wählen.

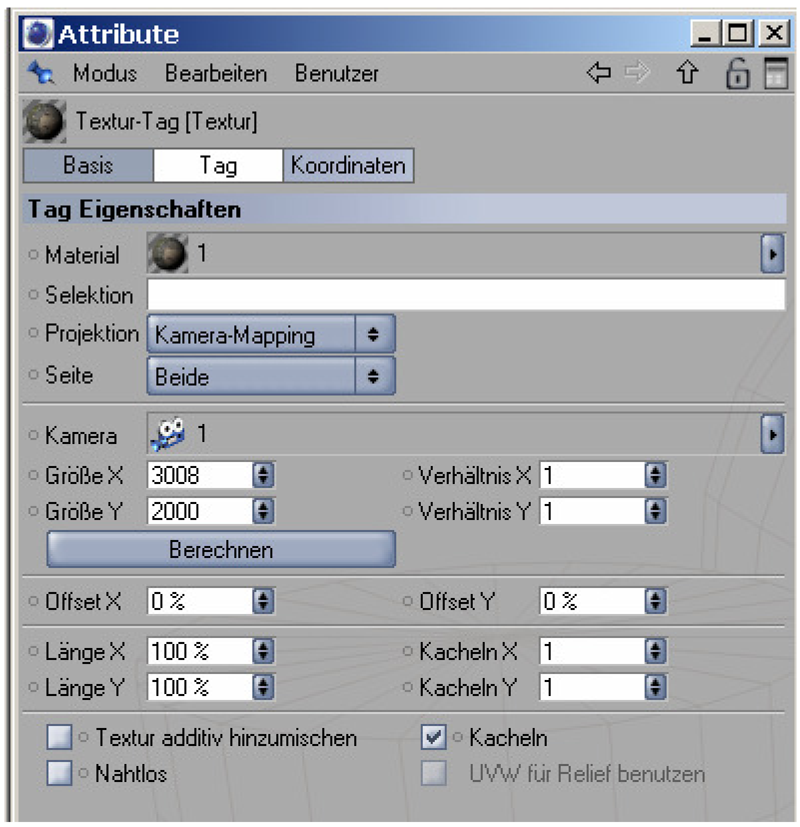


Abbildung : Tag Eigenschaften

Es sind die folgenden Einstellungen zu machen:

Projektion: Kamera-Mapping

Kamera: Die Kamera ist per Drag&Drop aus dem Objekt-Manager einzufügen.

Grösse X, Y: Hier sind die Abmessungen des Fotos in Pixel einzugeben.

Offset X, Y: Oft muss hier noch ein Offset gesetzt werden, so dass die Textur optimal auf das Objekt passt.

3. Polygon Anpassungen

Je nachdem müssen jetzt die einzelnen Teilobjekte an die Texturen angepasst werden. Vielleicht werden einzelne Polygone oder Teile des Objektes besser von einem anderen Foto texturiert? Für diesen Arbeitsschritt verwendet man am besten die folgenden Funktionen:

Abtrennen
Trennt die markierten Polygone vom Objekt ab und erzeugt aus ihnen ein neues Objekt.

Diesen Befehl findet man im Polygonbearbeitungsmodus in der 3D-Ansicht mit einem Rechtsklick. Die Polygone müssen dafür selektiert sein.

Verbinden
Fügt die im Objekt-Manager markierten Objekte zu einem neuen Objekt zusammen.

Sobald zwei oder mehr Objekte im Objekt-Manager markiert sind, kann diese Funktion mit Rechtsklick aufgerufen werden.

Damit können Polygone relativ einfach einem anderen Objekt zugewiesen werden. Wichtig ist, dass die Polygone nur einmal existieren. Also nachdem sie einem neuen Objekt zugewiesen wurden, müssen sie beim alten gelöscht werden.

12.2.2 Radiometrischen Korrekturen

Verfasser: D. Emmenegger

Der Ablauf, wie man mehrere Bilder radiometrisch abgleicht wird hier erläutert.

Grundlagen

- Es werden die verzeichnungsfreien Bilder aus PHIDIAS benötigt. Sie müssen bereits eindeutig nummeriert sein.
- Die Leica Photogrammetry Suite (LPS) muss vorhanden und bekannt sein.

Ablauf

Zuerst werden die Bilder für den radiometrischen Abgleich in den Image Equalizer der LPS eingelesen.

1. **File**
   Add Images

Anschliessen müssen alle Bilder selektiert werden, damit die Bildstatistiken gerechnet werden können.

2. **Edit**
   Select All

Die Bildstatistiken werden mit dem folgenden Befehl berechnet:

Image
Build Statistic Tables…

Nachdem alle Bildstatistiken berechnet wurden, kann der radiometrische Abgleich vorgenommen werden.

3. **Tools**
   Equalize…

Dann erscheint folgende Oberfläche:

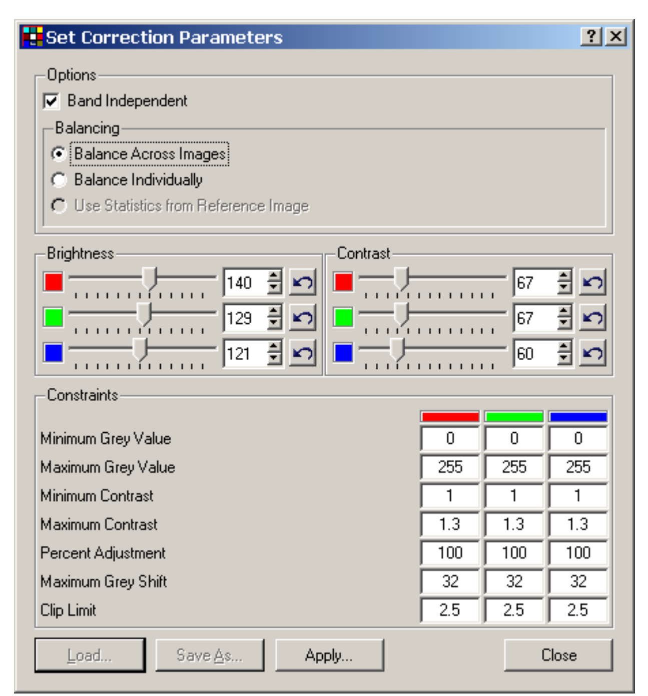


Abbildung : Equalize Tool in LPS

Mit diesen automatischen Einstellungen wird ein allgemeiner radiometrischer Abgleich über alle selektierten Bilder berechnet. Somit kann mit Apply fortgefahren werden. Es wird dieses Fenster angezeigt:

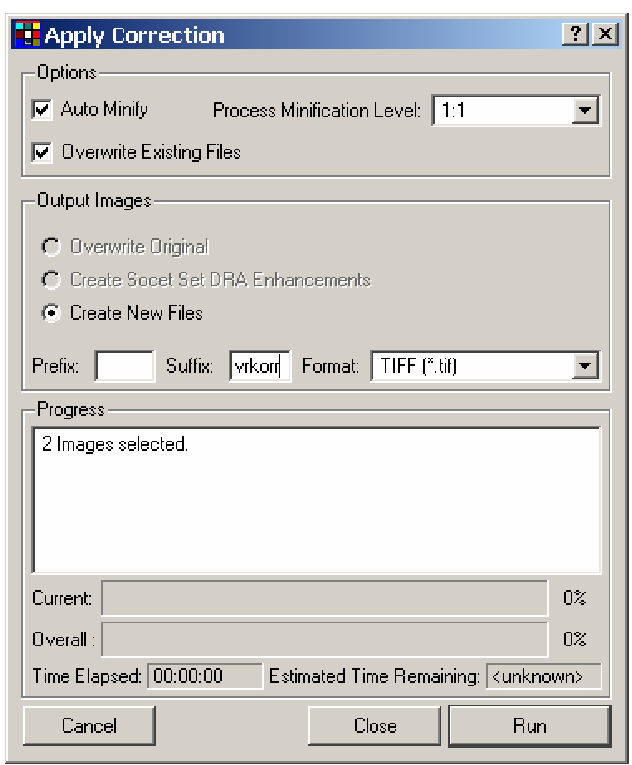


Abbildung : Ausgabe Einstellungen

Hier können einige Einstellungen zur Ausgabe gemacht werden. Wichtig dabei sind folgende Punkte:

- Ein Suffix sollte vergeben werden, damit die Bilder von den nicht korrigierten unterschieden werden können.

- Das beste Ausgabeformat ist ein TIF-File. JPEG-Files können nicht direkt ausgegeben werden.

Anschliessend kann der radiometrische Abgleich ausgeführt werden. Die Bilder sind jetzt zwar korrigiert, nur ist das TIF-Format für Cinema 4D nicht lesbar. Deswegen ist es nötig ein Export in ein JPEG-Format zu machen. Dafür verwenden wir die Export/Importfunktion in LPS.

4. Main
Import/Export…

Dabei wird folgendes Fenster aufgerufen:

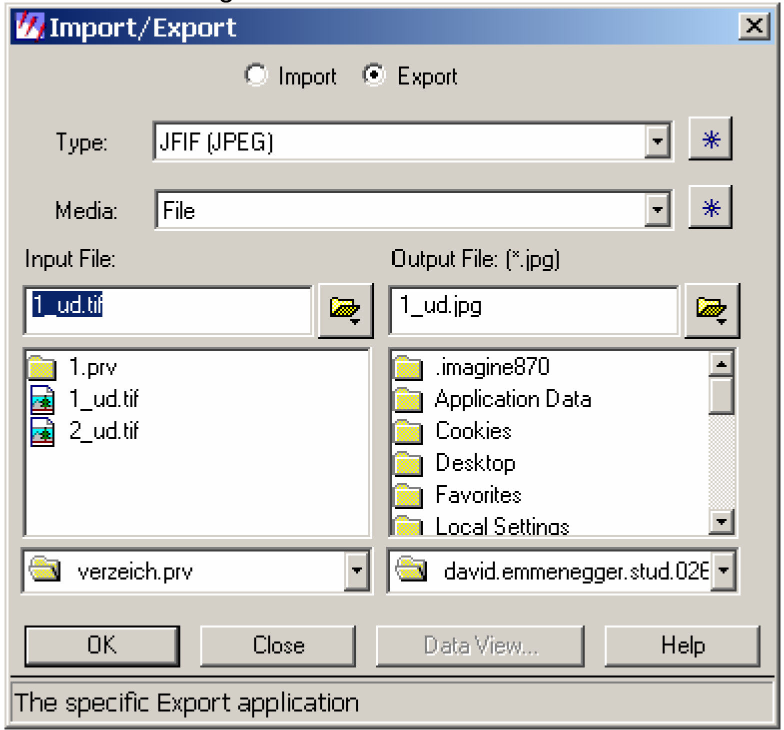


Abbildung : Import/Exportfunktion in LPS

Diese Funktion sollte selbsterklärend sein. Somit sind die Bilder nun bereit um in Cinema 4D texturiert zu werden.

12.2.3 Orthophoto einfügen

Verfasser: D. Emmenegger

Hier wird gezeigt, wie man ein Orthophoto kombiniert mit einen Geländemodell in eine Cinema 4D Szene integriert.

Grundlagen

- Orthophoto zugeschnitten auf den Geländemodellperimeter
- Geländemodell in einem DXF-File

Ablauf

Zuerst wird das Geländemodell in Cinema 4D importiert. Dafür müssen wir uns in Cinema 4D befinden.

1. Datei
Hinzuladen…
das Geländemodell anwählen

Das Geländemodell wird nun in Cinema 4D importiert und erscheint in der 3D-Ansicht. Anschliessend muss für die Texturierung des Geländemodells ein Material angelegt werden, welches das Orthophoto beinhaltet.

2. Material-Manager
Datei
Neues Material

Im Attribut-Manager kann nun das Orthophoto als Textur zugewiesen werden.

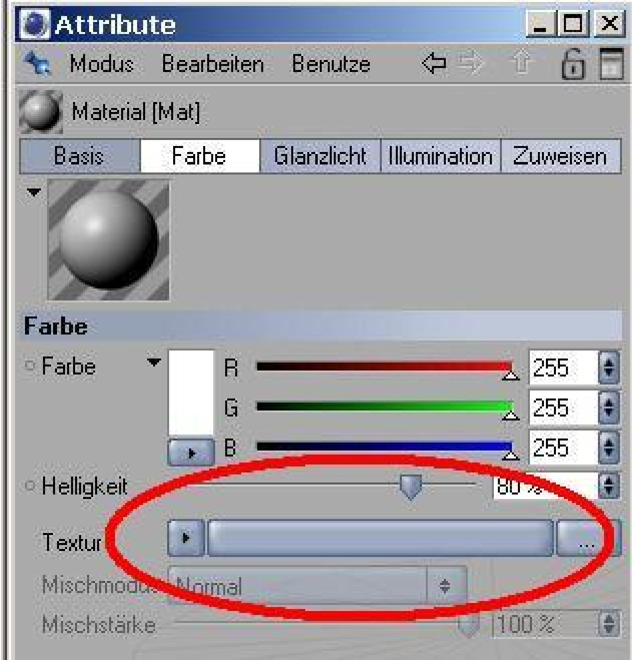


Abbildung : Material im Attribut-Manager

Zu überprüfen ist noch, dass unter der Registerkarte "Basis" die Option "Glanzlicht" deaktiviert ist. Ansonst wird das Orthophoto mit einem unnatürlichen Glanzlicht dargestellt. Anschliessend kann durch einfaches Drag&Drop das Material dem Geländemodell im Objekt-Manager zugewiesen werden.

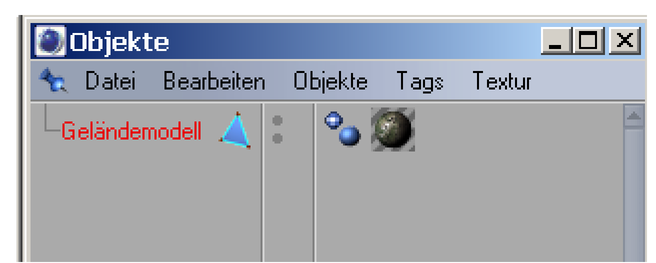


Abbildung : Objekt- Manager mit Objekt, Material und Kamera

Bei einem Doppelklick auf das Material im Objektmanager kommt man zu den "Textur-Tag" Eigenschaften des Materials. Dort kann man die Projektionseigenschaften des Materials wählen.

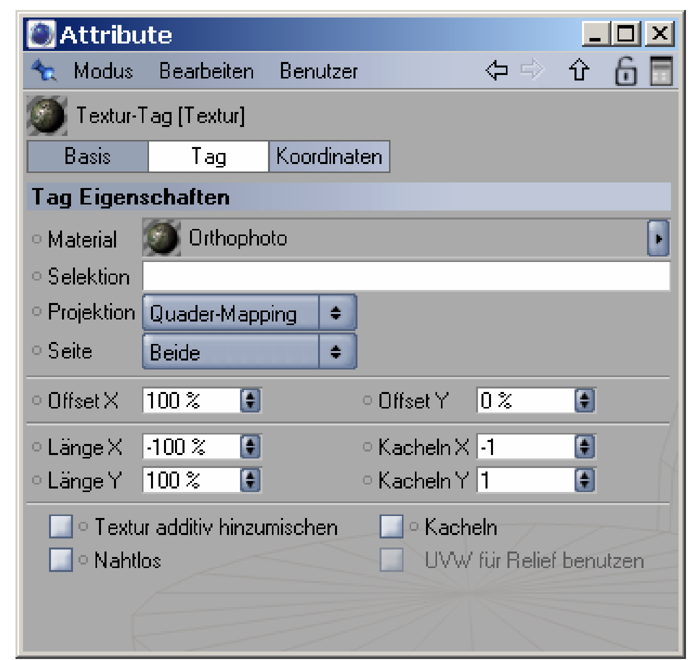


Abbildung : Tag Eigenschaften

Es ist nur die Projektion zu ändern:

Projektion: Quader-Mapping

Jetzt ist zwar das Orthophoto auf dem Geländemodell, aber die Skalierung stimmt noch nicht. Um diese anzupassen verwenden wir einen Textur-Tag im Objekt-Manager.

3. Textur
Auf Objekt Anpassen

Es kann sein, dass das Orthophoto noch gespiegelt werden muss. Auch diese Funktion ist unter dem Textur Menü zu finden.

12.2.4 Springbrunnen

Verfasser Th. Schuler

1) Aufrufen eines Partikel-GeometrieObjekt im Plugins-Menü von Cinema 4D unter Thinkings-Particels

2) Eine XPresso Expression im Objekt-Manger über das Menü Datei / Cinema 4D Tags dem Partikel-Geometrie-Objekt zuweisen.
   Jedes benötigte „Thinking Particles“Setup muss über den XpressoEditor erzeugt werden. Dieser wiederum muss einem beliebigen Objekt über einen XPresso-ExpressionTag zuerst zu-gewiesen werden. Für derartige Zwecke empfiehlt sich in den meisten Fällen ein Null-Objekt, da dies einerseits alle benötigten Informationen, wie z.B. eine Position und Ausrichtung im Raum liefern kann, anderseits aber beim Rendern unsichtbar bleibt. (Manuel Thinking Particles S.98). Diese Eigenschaften hat auch ein Partikel-Geometrie-Objekt.

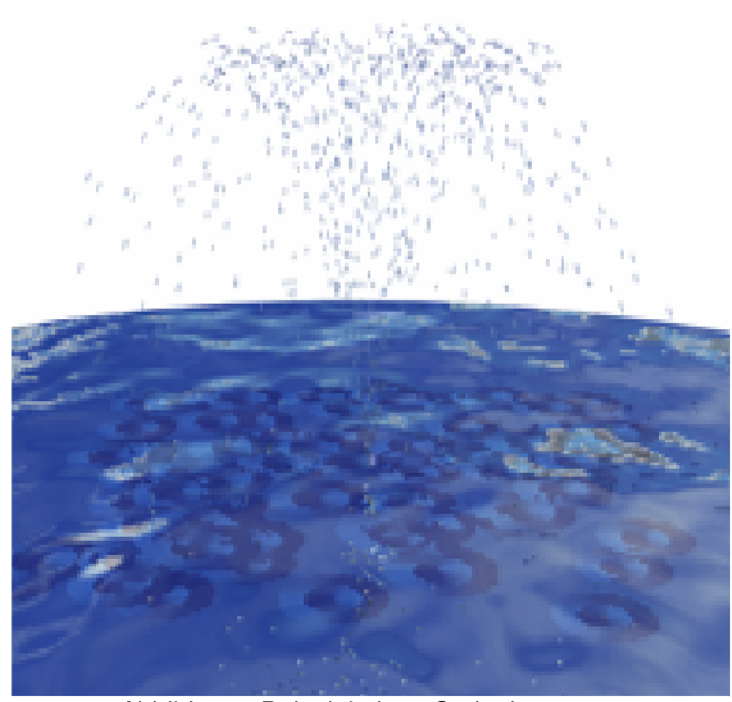


Abbildung : Beispiel eines Springbrunnens

3) Im Kontext-Menü des XPresso-Editor durch Rechtsklick der Maus unter Neuer Node / Thinking Particles / TP Generator eine P Sturm-Node wählen. Dem bereits vorhandenen Emitter Position-Eingang-Port am P Sturm-Node noch den Eingang-Port Emitter Ausrichtung hinzufügen. Der Eingang-Port ist mit einem Klick auf die blaue Schaltfläche in der Titelzeile des Nodes abrufbar.

4) Das Partikel-Geometrie-Objekt aus dem Objekt-Manager in den XPresso-Editor hineinziehen. Es entsteht dadurch ein Objekt-Node für das Partikel-Geometrie-Objekt.

5) Am Partikel-Geometrie-Objekt den Ausgangs-Port für die Globale Position und die Globale Matrix aktivieren.
   Da der P Sturm-Emitter an sich kein Objekt ist, das in den Editor-Ansichten verschoben oder verdreht werden kann, muss der Node an die Position und Ausrichtungen des Partikel-Geometrie-Objektes angeschlossen werden. Diese Informationen erhält er z.B. aus der globalen Position und der globalen Matrix eines Objekts. Ab dem Zeitpunkt der Verknüpfung gilt, dass sich der Emitter durch

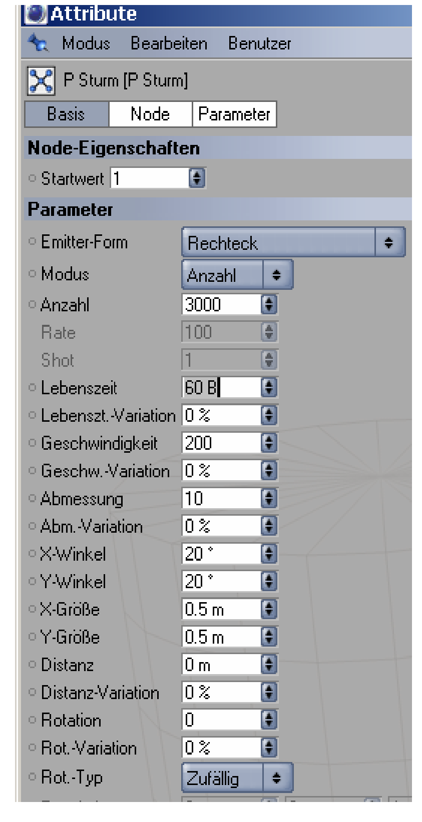


Abbildung : PSturm Attribute-Manager

Verschieben oder Rotieren des Null-Objekts manipulieren lässt. Die Z-Achse des Null-Objekts fungiert dabei als Ausstossrichtung für die Partikel (Manuel Thinking Particles S.100).

6) Selektieren des P-Sturm-Node, um dessen Parameter auf der Parameter-Seite des Attribute-Managers angezeigt zu bekommen. Hier müssen verschiedene Einstellung betätigt werden. Jeder einzelne Wert verändert das Erscheinungsbild stark und muss den Gegebenheiten (Grösse des Objektes) angepasst werden. Die einzelnen Erklärungen zu den Schaltflächen befinden sich im Thinking Partikels Manuel auf Seite 42.

7) Dem P Sturm-Node – durch Anklicken der roten Schaltfläche in der Titelzeile des Nodes – einen Ausgang-Port für Geborene Partikel hinzufügen.

8) Über das Kontext-Menü Neuer Node / Thinking Particles / TP Standard einen P Gruppe-Node erzeugen.

9) Aktivieren des P Gruppe-Node im XPresso-Editor durch einen Mausklick auf die Titlezeile des Nodes. Im Attribute-Manger wird daraufhin eine „Drag and Drop"-Zeile für die Partikel-Gruppe sichtbar. Aufrufen der Thinking Paricles-Einstellungen unter Plugins /TinkingParticles und dort die Partikel-Gruppe Alle in die „Drag and Drop"-Zeile hineinziehen.
   Partikel-Gruppen kommen im Thinking Paricles-System eine entscheidende Rolle zu, denn über die Zugehörigkeit von Partikeln zu einer Gruppe kann das Verhalten dieser Partikel von andern Partikeln abgegrenzt werden (Manuel Thinking Particles S.102). Dies ist hier nicht nötig, darum wurde auch keine neue Gruppe erzeugt.

10) Abrufen eines P Pass-Node aus dem XPresso-Editor-Kontext-Menü unter Neuer Node / Thinking Particles / TP Initiator. Den Partikelname aus den Thinking Particles-Einstellungen in das Gruppe-Feld im Attribute-Manager ziehen.

11) Erstellen eines Würfels-Objekts.
   Der Würfel wird für den zukünftigen P-Gravitation-Node gebraucht. Eine Gravitation wirkt immer in die Z-Richtung, deshalb muss der Würfel in die gewünschte Richtung gedreht werden.

12) Aufrufen eines P Gravitation-Node aus dem Kontext-Menü Neuer Node / Thinking Particles / TP Dynamik. Den Würfel aus dem Objekt-Manager in das Objekt-Feld des P Gravitation-Nodes im Attribute-Manager ziehen. Die wichtigste Einstellung der P Gravitation ist sicherlich die Stärke, welche den Gegebenheiten angepasst werden muss. Die Beschreibungen zu den Einstellungen der P Gravitation findet man im Thinking Particels Manuel auf Seite 78. Verbinden des P Pass-Nodes mit dem Partikel-Eingang des P Gravitation-Node.
   Damit die erzeugten Partikel manipuliert werden können, müssen die Partikel über einen P Pass-Node abgerufen werden. Der P Sturm-Node hat zwar auch einen Ausgang für Partikel, dieser

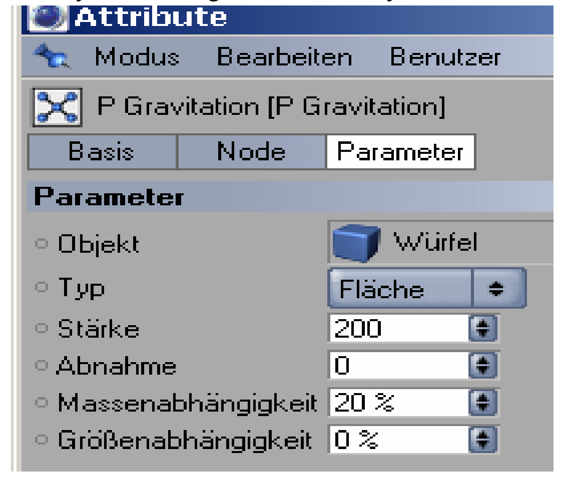


Abbildung : Gravitaions Attribute-Manager

liefert jedoch nicht die Daten aller Partikel, sondern nur die Daten der gerade zu dem Zeitpunkt neu erzeugten / geborenen Partikel. Das Einleiten des P Pass-Partikelstroms in einen P Gravitation-Node erlaubt es, die Partikel mit einer Kraft zu beeinflussen, die in diesem Fall ein Würfel von der Position eines Objekts ausgeht (Manuel Thinking Particles S.105).

13) Um den Partikel eine Form zu geben, muss ein Polygon-Objekt verwendet werden. Dabei ist am besten ein Grundobjekt aus dem Cinema-4D-Menü zu wählen. Dieses muss den Gegebenheiten und eigenen Wünschen angepasst werden. Hier wurde eine Kapsel genommen.

14) Abrufen eines P Form-Node aus dem Kontext-Menü Neuer Node / Thinking Particles / TP Standard. Bei aktivem P Form-Node die Kapsel aus dem Objekt-Manager in das Objekt-Feld im Attribute-Manager ziehen. Einen Umkugel-Radius von „10“ wählen. Siehe Erklärungen zu Einstellungen des P Form im Thinking Particles Manuel auf Seite 50.

15) Verbinden des P Pass Ausgangs mit dem P Form Eingang
Über einen P-Form-Node kann ein Objekt an die Partikel weitergeleitet werden. Da in diesem Fall der P Form-Node direkt auf alle Partikel des Emitters zugreifen kann, wird jeder Partikel durch die Kapsel ersetzt (Manuel Thinking Paricels S.104)[7].

16) Aufrufen eines P Ausrichtungs-Node aus dem Kontext-Menü Neuer Node / Thinking Particles / TP Standard.

17) Verbinden des P Pass Ausgangs mit dem P Ausrichtung Eingang.
Die Ausrichtung der Partikel während ihres Fluges ist von extremer Wichtigkeit. Um automatisch zu sorgen, das die Kapsel nicht einfach senkrecht rauf und wieder runter gehen muss diese Ausrichtung vorgenommen werden. In diesem Beispiel sind die Grundeinstellungen des P Ausrichtung-Node gerade richtig eingestellt. Nähere Angaben zu den Befehlsfeldern siehe Manuel Seite 53.

Der XPresso-Editor sollte nun so aussehen:

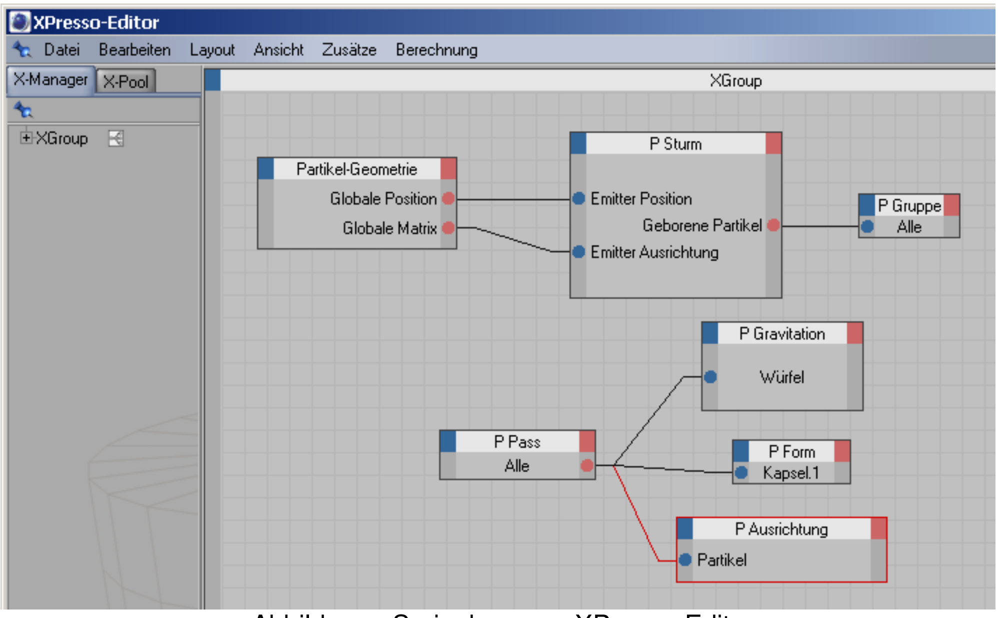


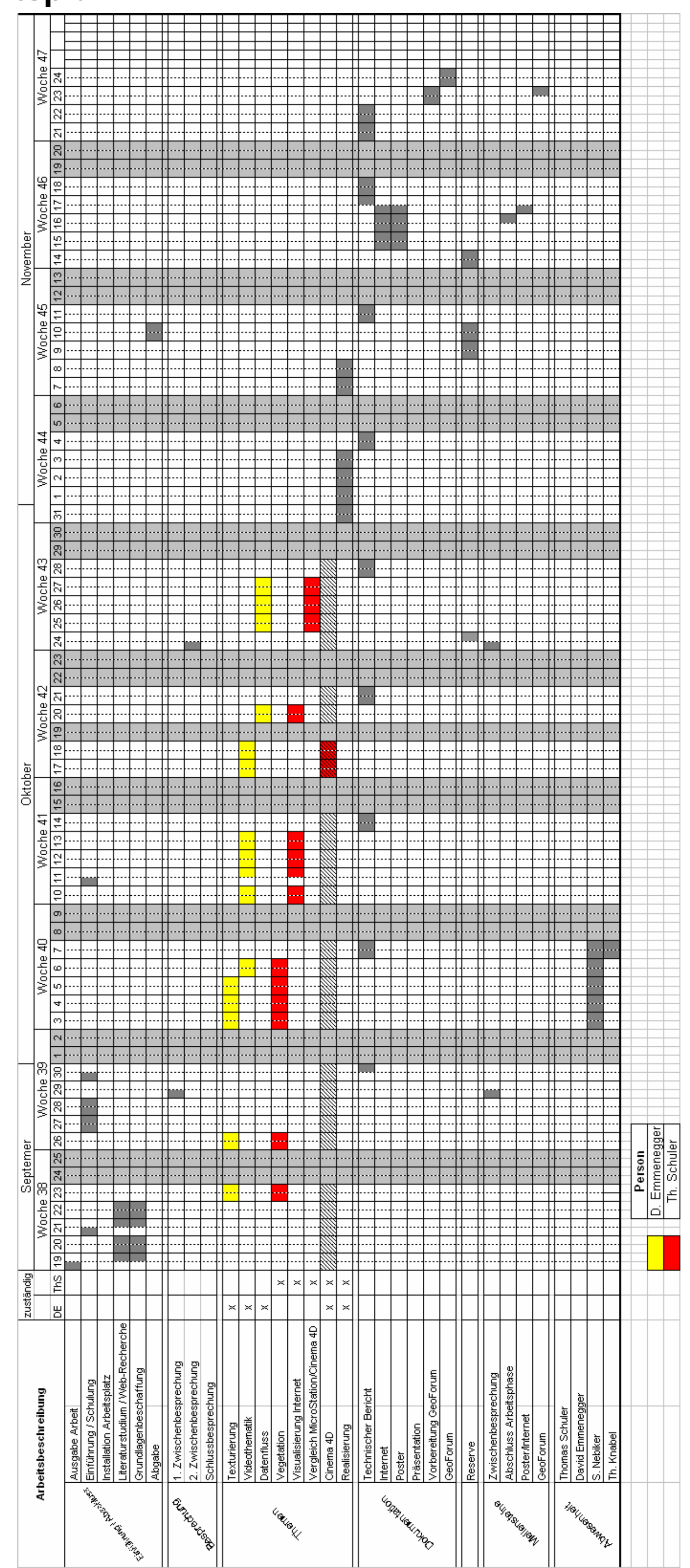
Abbildung : Springbrunnen XPresso-Editor

18) Im Material-Manger ein über Datei ein Neues Material erstellen. Durch Doppelklick auf das neu erstellte Material Symbol im Material-Manger kommt man in den Material-Editor. Dort in der Option Relief bei Textur auf den nebenstehende waagrechten Peil drücken. Nun kann unter Oberfläche Wasser ausgewählt werden. Es steht jetzt im Relief unter Textur „Wasser". Durch Doppelklick kommt man zu den Eigentlichen Eigenschaften des WasserShaders. Beschreibung zu den einzelnen Einstellungen siehe Referenzhandbuch S.768. Die Einstellungen des WasserShaders sowie des Material-Editor sind den Wünschen anzupassen. Grundsätzlich erhält man ein gutes Resultat, wenn bei der Transparenz etwa eine Helligkeit von ca. 20% herrscht und bei der Spiegelung die Helligkeit um 80% gesetzt wird.

19) Dem gleichen Objekt muss nun noch ein Wellen-Shader zugewiesen werden. Dabei genau gleich vorgehen. Ein neues Material erstelllen und den Shader, welcher unter Effekte / Wasser zu finden ist wieder im Relief-Kanal platzieren. Die Erläuterungen zu den Welleneigenschaften findet man im Referenzhandbuch auf der Seite 755. Wichtig ist dabei, dass unter den Shader-Eigenschaften durch „Drag & Drop" das richtige Objekt auf welche die Wellen dargestellt werden soll und die Partikel aus dem Objekte-Manager hineingezogen werden.

12.3 Projektmanagement

12.3.1 Arbeitsplan



12.3.2 Projektjournale

12.3.2.1 Thomas Schuler

19.September 2005	Arbeiten	Arbeitszeit
	• Einführung / Auftragsbesprechung • Web-Recherechen; cinema4d • Arbeitsplanung / Projektmagenement • Installation cinema4d	
20.September 2005	**Arbeiten**	**Arbeitszeit**
	• Vertrautmachen mit Cinema4d • Berichtaufbau (grob)/Zeitplanung	
21.September 2005	**Arbeiten**	**Arbeitszeit**
	• Einfügen von Homburg in Cinema4d • Einführung in G-Vista • (fühl mich heute beschissen, Kopfschmerzen)	Einfügen: 0.5h
22.September 2005	**Arbeiten**	**Arbeitszeit**
	• Lit.studium über Möglichkeiten Pflanzen virtuell darzustellen • Bericht: Vegetation • MicroStation-Fähigkeiten auffrischen	
23.September 2005	**Arbeiten**	**Arbeitszeit**
	• Übersetzen zweier Texte vom Englischen ins Deutsch	
24.September 2005	**Arbeiten**	**Arbeitszeit**
25.September 2005	**Arbeiten**	**Arbeitszeit**
26.September 2005	**Arbeiten**	**Arbeitszeit**
	• Demo DILAS mit Knabel • Vorbereitung Sitzung 1 • Cinema 4D R9.5 installieren	
27.September 2005	**Arbeiten**	**Arbeitszeit**
	• Cinema4D R9.5 Einführung + eigene Tests	**9h**
28.September 2005	**Arbeiten**	**Arbeitszeit**
	• Cinema4D R9.5 Einführung + eigene Tests	**9.5h**
29.September 2005	**Arbeiten**	**Arbeitszeit**
	• Vorbereitung 1.ZWB • 1.ZWB • Protokoll schreiben • 2. Bildschirm installieren • Cyclone installieren	
30.September 2005	**Arbeiten**	**Arbeitszeit**
	• G-Vista Einführung • Punktwolke in Cyclone zuschneiden • Homburg in Cinema4d importieren • Bäume in Cinema4d importieren • Bericht; Vegetation • Datensicherheitskonzept ergänzen	
1. Oktober	**Arbeiten**	**Arbeitszeit**
2. Oktober	**Arbeiten**	**Arbeitszeit**

3. Oktober	**Arbeiten**	**Arbeitszeit**
	• Einarbeitung in Thema Partikel + Übungen/Beispiele erarbeiten	**8.5h**
4. Oktober	**Arbeiten**	**Arbeitszeit**
	• Partikelübungen • Punktwolke in Cinema4D einführen und experimentieren über Partikel	3.5h 5h **8.5h**
5. Oktober	**Arbeiten**	**Arbeitszeit**
	• Aus Punktwolke in Cinema4D versuchen Baum zu modellieren über Polygone • Hair Department herunterladen und experimentieren für Grasherstellung • Partikel-Windanimation	2.5h 6h 1h **9.5h**
6. Oktober	**Arbeiten**	**Arbeitszeit**
	• Hair Department Animation • XPresso • Bericht Grundlagen Vegetation • Cinema 4D allg. (z.B. Partikel;Rauch/Wasser)	4h 1.5h 0.5h 3h **9h**
7. Oktober	**Arbeiten**	**Arbeitszeit**
	• Mit Baumbibi experimentieren; spezifische Änderungen vornehmen an der Form eines Baumes	**7h**
8. Oktober	**Arbeiten**	**Arbeitszeit**
9. Oktober	**Arbeiten**	**Arbeitszeit**
10. Oktober	**Arbeiten**	**Arbeitszeit**
	• G-Vista	**8.5h**
11. Oktober	**Arbeiten**	**Arbeitszeit**
	• G-Vista (Javascript-Einführung mit Pascal) • G-Vista vrml-Import • Besprech. mit Herr Nebiker • Buch „3D Visualisierung lesen“	
12. Oktober	**Arbeiten**	**Arbeitszeit**
	• Springbrunnen entwickeln	**9h**
13. Oktober	**Arbeiten**	**Arbeitszeit**
	• Springbrunnen- / Feuerenwicklung	**8.5h**
14. Oktober	**Arbeiten**	**Arbeitszeit**
	• Feuerentwicklung	**7h**
15. Oktober	**Arbeiten**	**Arbeitszeit**
16. Oktober	**Arbeiten**	**Arbeitszeit**
17. Oktober	**Arbeiten**	**Arbeitszeit**
	• Technischer Bericht	**9h**
18. Oktober	**Arbeiten**	**Arbeitszeit**
	• Tech. Bericht • Feuerentwicklung; PyroCluster • Vorbereitung 3DGI	3h 5.5h 0.5h **9h**
19. Oktober	**Arbeiten**	**Arbeitszeit**
	• Feuerentwicklung; PyroCluster • Metaball(für Laserscanning-Bäume) • 3DGI	2h 2.5h 5.5h **10h**

20. Oktober	Arbeiten	Arbeitszeit
	• Metaball(für Laserscanning-Bäume) • Atom-Array (für Laserscanning-Bäume) • Vorbereitung 2.Zwischenbesprechung • Tech. Bericht	1h 6.5h 0.75h 0.75h **9h**
21. Oktober	**Arbeiten**	**Arbeitszeit**
	• Neuer Baum(Cyclone/Cinema4d) • G-Vista • Vorbereitung 2.Zwischenbesprechung • Tech. Bericht	2.75h 3.0h 0.75h 2.5h **9h**
22. Oktober	**Arbeiten**	**Arbeitszeit**
23. Oktober	**Arbeiten**	**Arbeitszeit**
24. Oktober	**Arbeiten**	**Arbeitszeit**
	• Tech. Bericht • Vorbereitung 2. Zwischenbesprechung • 2. Zwischenbesprechung • Protokoll 2.ZWB	4.25h 0.5h 1.75h 1.75h **8.25h**
25. Oktober	**Arbeiten**	**Arbeitszeit**
	• Feuerentwicklung • Homburg Grundlagen erarbeiten	3.75h 4.25h **8h**
26. Oktober	**Arbeiten**	**Arbeitszeit**
	• Homburg Grundlagen erarbeiten • Baum-Punktwolke in Cyclone erstellen	5h 3h **8h**
27. Oktober	**Arbeiten**	**Arbeitszeit**
	• Homburg Grundlagen • G-Vista • Cyclone in Kombi vbexteriors	3.5h 0.5h 4.5h **8.5h**
28. Oktober	**Arbeiten**	**Arbeitszeit**
	• Bericht • Cyclone in Kombi vbexteriors an Homburg	2.5h 5.5h **8h**
29. Oktober	**Arbeiten**	**Arbeitszeit**
	• Realisierung Homburg	**6h**
30. Oktober	**Arbeiten**	**Arbeitszeit**
31. Oktober	**Arbeiten**	**Arbeitszeit**
	• Realisierung Homburg	**10h**
1. November	**Arbeiten**	**Arbeitszeit**
	• Realisierung Homburg • Besprechung mit HGK	10h 1.5h **11.5h**
2. November	**Arbeiten**	**Arbeitszeit**
	• G-Vista/3DStudio • Realisierung Wildenstein	5.25h 4.75h **10h**
3. November	**Arbeiten**	**Arbeitszeit**
	• G-Vista/3DStudio • Realisierung Wildenstein	5h 5h **10h**

4. November	**Arbeiten**	**Arbeitszeit**
	• Realisierung Wildenstein • Tech. Bericht • G-Vista	1h 3h 5.5h **9.5h**
5. November	**Arbeiten**	**Arbeitszeit**
6. November	**Arbeiten**	**Arbeitszeit**
7. November	**Arbeiten**	**Arbeitszeit**
	• Tech. Bericht • G-Vista	3h 7h **10h**
8. November	**Arbeiten**	**Arbeitszeit**
	• Bericht	**10.5h**
9. November	**Arbeiten**	**Arbeitszeit**
	• Bericht • Besprechung mit HGK • G-Vista	4.5h 3.75h 0.75h **9.0h**
10. November	**Arbeiten**	**Arbeitszeit**
	• G-Vista • Bericht	3h 6.5h **9.5h**
11. November	**Arbeiten**	**Arbeitszeit**
	• G-Vista • Bericht	2.0h 7.0h **9.0h**
12. November	**Arbeiten**	**Arbeitszeit**
	• Bericht	**5h**
13. November	**Arbeiten**	**Arbeitszeit**
14. November	**Arbeiten**	**Arbeitszeit**
	• Poster / Internet • Bericht	6.5h 5h **11.5h**
15. November	**Arbeiten**	**Arbeitszeit**
	• Bericht • Poster / Internet • G-Vista	9.0h 1.0h 2.0h **12.0h**
16. November	**Arbeiten**	**Arbeitszeit**
	• Bericht • Poster / Internet	8.5h 1.0h **9.5h**
17. November	**Arbeiten**	**Arbeitszeit**
	• Bericht	**11h**
18. November	**Arbeiten**	**Arbeitszeit**
	• Bericht • Präsentation	5.5h 1.5h **7h**
19. November	**Arbeiten**	**Arbeitszeit**
	• Bericht	**2h**
20. November	**Arbeiten**	**Arbeitszeit**
	• Bericht	**8.0h**
21. November	**Arbeiten**	**Arbeitszeit**
	• Bericht • Präsentation überarbeiten	6.0h 4.0h

		10.0h
22. November	**Arbeiten**	**Arbeitszeit**
	• Bericht überarbeiten und Abschlussarbeiten • Präsentation GeoForum schreiben + vorbereiten	4.0h 4.0h **8.0h**
23. November	**Arbeiten**	**Arbeitszeit**
	• GeoForums Präsentationen vorbereiten	**8.0h**
24. November	**Arbeiten**	**Arbeitszeit**
	• GeoForum Vorbereitung • GeoForum	4.0h 8.0h **12.0h**
25. November	**Arbeiten**	**Arbeitszeit**
	• Abschlussarbeiten	**4.0h**

12.3.2.2 David Emmenegger

19.September 2005	**Arbeiten**	**Arbeitszeit**
	• Einführung durch Gottwald / Besprechung mit Nebiker, Knabl und Schuler • Datensicherungs / Datenorganisationskonzept • inkl. Anlegen der Verzeichnisse • Web- Recherche über Cinema 4D • Literaturstudium (im Zug)	3.5h 3.0h 1.0h 0.5h 0.5h **8.5h**
20.September 2005	**Arbeiten**	**Arbeitszeit**
	• Literaturstudium "Graphische Datenverarbeitung" (Texturen) • Installation Software (IMO, Adobe Premiere) und Bereitstellung der Dokumentationen • Zeitplanung und provisorisches Inhaltsverzeichnis • Ausgangslage - Datenfluss	1.5h 2.0h 1.0h 4.0h **8.5h**
21.September 2005	**Arbeiten**	**Arbeitszeit**
	• Einführung in G-Vista Suite und anschliessenden Test • Vorbereitung Besprechung • Ausgangslage - Datenfluss • Bericht (Motivation)	4.5h 0.5h 3.0h 0.5h **8.5h**
22.September 2005	**Arbeiten**	**Arbeitszeit**
	• Vorbereitung und Besprechung (Nebiker, Knabl, Schuler) • MicroStation (Animation, Textur / suche nach Forums / Turtorials)	1.5h 9.0h **10.5**
23.September 2005	**Arbeiten**	**Arbeitszeit**
	Theoretische Grundlagen Texturierung erarbeiten	8.0h **8.0h**
24.September 2005	**Arbeiten**	**Arbeitszeit**
25.September 2005	**Arbeiten**	**Arbeitszeit**
	• Lesen des Berichtes " Visualisation of Highly Textured Surfaces"	1.5h **1.5h**

26.September 2005	Arbeiten	Arbeitszeit
	• Vorbereitung Sitzung 1 • Demo DILAS mit T. Knabl • Texturierung Ablauf • Vorbereitung Cinema 4D Schulung, inkl. Installation	3.0h 2.0h 2.0h 2.5h **9.5h**
27.September 2005	**Arbeiten**	**Arbeitszeit**
	• Einführung Cinema 4D und eigene Versuche	10.0h **10.0h**
28.September 2005	**Arbeiten**	**Arbeitszeit**
	• Einführung Cinema 4D und eigene Versuche	10.0h **10.0h**
29.September 2005	**Arbeiten**	**Arbeitszeit**
	• Vorbereitung + Sitzung 1 • Eigene Szenen im Cinema 4D	2.0h 7.0h **9.0h**
30.September 2005	**Arbeiten**	**Arbeitszeit**
	• Einführung G-Vista Composer • Import von VRML Dateien in Cinema 4D • Texturierung mit Kameraprojektion	2.0h 3.0h 4.0h **9.0h**
1.Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Literaturstudium: o Virtualizing a Byzantine Crypt… o Automatic Multi-Image Photo-Texturing of 3D Surface Models…	1.0h 1.0h **2.0h**
2.Oktober 2005	**Arbeiten**	**Arbeitszeit**

3.Oktober 2005	Arbeiten	Arbeitszeit
	• Nachlieferung von Grundlagedaten • Literaturstudium • Theoretische Grundlagen Texturierung, Bericht schreiben	1.0h 0.5h 7.5h **9.0h**
4. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Theoretische Grundlagen Texturierung, Bericht schreiben	9.0h **9.0h**
5. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Theoretische Grundlagen Texturierung, Bericht schreiben • Texturierung mit Kameramapping	2.0h 8.0h **10.0h**
6. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Texturierung mit Kameramapping • Export dgn -> dxf	2.0h 7.0h **9.0h**
7. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Texturierung mit Kameramapping • Exportmöglichkeiten Testen	4.0h 5.0h **9.0h**
8. Oktober 2005	**Arbeiten**	**Arbeitszeit**
9.Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Literaturstudium: o 3-D Reconstruction of complex Architectures … o 3D modelling with reusable and integrated…	1.0h 1.0h **2.0h**

10.Oktober 2005	Arbeiten	Arbeitszeit
	• Installation und Einarbeitung in NetRender • Renderung von einer Beispielszene • Einarbeitung in Adobe Premiere Pro	2.0h 4.0h 4.0h **10.0h**
11. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Abklärung zu VRML Export aus Cinema4D • Einarbeitung in Videothematik • Besprechung mit S. Nebiker	5.0h 4.5h 1.0h **10.5h**
12. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Literaturstudium o 3D Visualisierung	 3.0h **3.0h**
13. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Orthophoto in Cinema 4D • Phidias-Winkel Umrechnung • Verzeichnungsberechnung der Bilder	4.0h 4.0h 3.0h **11.0h**
14. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Phidias-Winkel Umrechnung • PTLens	4.5h 4.5h **9.0h**
15. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Internetrecherche über PTLens	2.0h **2.0h**
16.Oktober 2005	**Arbeiten**	**Arbeitszeit**

17.Oktober 2005	Arbeiten	Arbeitszeit
	• Texturierung von Wildenstein • Panotools	8.0h 2.0h **10.0h**
18. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Texturierung von Wildenstein • Film erstellen für 3DGI	8.5h 3.0h **11.5h**
19. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• 3DGI	
20. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• PHIDIAS-Update • Vorbereitung Sitzung 2 • Bericht	3.0h 3.0h 2.5h **8.5h**
21. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Bericht • Vorbereitung Sitzung 2 • VRML Export	4.0h 4.0h 1.0h **9.0h**
22. Oktober 2005	**Arbeiten**	**Arbeitszeit**
23.Oktober 2005	**Arbeiten**	**Arbeitszeit**

24.Oktober 2005	Arbeiten	Arbeitszeit
	• Vorbereitung + Sitzung 2 • Bericht • Phidias-Verzeichnung	3.0h 4.0h 2.0h **9.0h**
25. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Phidias-Verzeichnung • Radiometrische Korrekturen • Bericht	4.0h 3.5h 1.5h **9.0hh**
26. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Abwesend	
27. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Phidias-Verzeichnung • Radiometrische Korrekturen • Bericht	2.5h 2.5h 5.0h **10.0h**
28. Oktober 2005	**Arbeiten**	**Arbeitszeit**
	• Bericht • Radiometrische Korrekturen • Grundlagen Wildenstein	3.0h 4.0h 3.0h **10.0h**
29. Oktober 2005	**Arbeiten**	**Arbeitszeit**
30.Oktober 2005	**Arbeiten**	**Arbeitszeit**

31.Oktober 2005	Arbeiten	Arbeitszeit
	• Grundlagen Wildenstein • Verwirklichung Wildenstein	4.0h 5.5h **9.5h**
1. November 2005	**Arbeiten**	**Arbeitszeit**
	• Verwirklichung Wildenstein • Besprechung	9.0h 1.5h **10.5h**
2. November 2005	**Arbeiten**	**Arbeitszeit**
	• Verwirklichung Wildenstein • Grundlagen Homburg	4.5h 4.5h **9.0h**
3. November 2005	**Arbeiten**	**Arbeitszeit**
	• Grundlagen Homburg	12.0h **12.0h**
4. November 2005	**Arbeiten**	**Arbeitszeit**
	• Net Renderer • Bericht	1.5h 7.0h **8.5h**
5. November 2005	**Arbeiten**	**Arbeitszeit**
6. November 2005	**Arbeiten**	**Arbeitszeit**

7 November 2005	Arbeiten	Arbeitszeit
	• Homburg • Net Renderer • Bericht	7.0h 2.5h 2.5h **12.0h**
8. November 2005	**Arbeiten**	**Arbeitszeit**
	• Homburg • Bericht	1.0h 9.0h **10.0h**
9. November 2005	**Arbeiten**	**Arbeitszeit**
	• Bericht • Besprechung HGK • Film Wildenstein	2.5h 3.0h 6.0h **11.5h**
10. November 2005	**Arbeiten**	**Arbeitszeit**
	• Film Wildenstein	9.0h **9.0h**
11. November 2005	**Arbeiten**	**Arbeitszeit**
	• Film Wildenstein • Film Homburg	6.0h 5.0h **11.0h**
12. November 2005	**Arbeiten**	**Arbeitszeit**
	• Film Wildenstein • Film Homburg	2.0h 3.0h **5.0h**
13. November 2005	**Arbeiten**	**Arbeitszeit**

14 November 2005	Arbeiten	Arbeitszeit
	• G-Vista Szene • Web-Zusammenfassung • Bericht	2.0h 4.0hh 3.5h **9.5h**
15. November 2005	**Arbeiten**	**Arbeitszeit**
	• Bericht	10.5h **10.5h**
16. November 2005	**Arbeiten**	**Arbeitszeit**
	• Abläufe • Bericht	2.5h 8.0h **10.5h**
17. November 2005	**Arbeiten**	**Arbeitszeit**
	• Abläufe • Bericht	5h 5h **10.0**
18. November 2005	**Arbeiten**	**Arbeitszeit**
	• Abläufe • Bericht (Korrekturen)	3.0h 5.0h **8.0h**
19. November 2005	**Arbeiten**	**Arbeitszeit**
20. November 2005	**Arbeiten**	**Arbeitszeit**
	• Bericht (Korrekturen)	4.0h **4.0h**
21 November 2005	**Arbeiten**	**Arbeitszeit**
	• Bericht • Abschlussarbeiten	3.0h 6.0hh **9.0h**
22. November 2005	**Arbeiten**	**Arbeitszeit**
	• Bericht drucken • GeoForum vorbereiten	4.0h 4.0h **8.0h**
23. November 2005	**Arbeiten**	**Arbeitszeit**
	• Geoforum vorbereiten • Abschlussarbeiten	6.0h 2.0h **8.0h**
24. November 2005	**Arbeiten**	**Arbeitszeit**
	• GeoForum Vorbereitung • GeoForum	4.0h 8.0h **12.0h**
25. November 2005	**Arbeiten**	**Arbeitszeit**
	• Abschlussarbeiten	4.0h **4.0h**

12.4 Datenmanagement

12.4.1 Datenorganisationskonzept

Im Diplomarbeitsverzeichnis "**N:\Diplomarbeiten\G2002\DA11**" wird folgende Verzeichnisstruktur angelegt:

- Arbeitsordner
- Austausch
- Datensicherung
- Diverses
- Dokumente
- Grundlagen
- Vorlagen

Die diversen Unterverzeichnisse werden hier nicht aufgeführt. Sie werden zum Teil erst noch nach Bedarf erstellt.

B. Lüscher S. Nebiker

…………………….. ……………………….

12.4.2 Datensichrungskonzept

Dieses Konzept soll Auskunft über das Datensicherungskonzept während der Diplomarbeit "Texturierung, Visulisierung und Animation komplexer 3D-Geoobjekte" geben. Die einzelnen Punkte und Verantwortlichkeiten werden im Folgenden dargelegt.

Gruppenmitglieder : Thomas Schuler
David Emmenegger

Verantwortliche Person : David Emmenegger

- **Wer ist für die Datensicherung Verantwortlich?**

Für die tägliche Sicherung der Daten auf dem Personalcomputer ist jeder Benutzer selbst Verantwortlich. Die Wöchentliche Sicherung wird durch die Verantwortliche Person der Gruppe durchgeführt.

Die Sicherheit für den Server N liegt bei der FHBB und wird mit einem alltäglichen Backup um 20.00 Uhr sowie einem Monatsbackup welches ausserhalb des Hauses aufbewahrt wird gewährleistet.

- **Wann wird die Datensicherung durchgeführt?**

Die tägliche Datensicherung wird jeweils bei Arbeitsschluss von jeder Person durchgeführt. Am Wochenschluss wird von der verantwortlichen Person die Wochensicherung vorgenommen.

- **Welche Daten werden überhaupt gesichert?**

Es werden Grundsätzlich die ganzen lokalen Verzeichnisse gesichert. Dies soll eine möglichst einfache und schnelle Sicherung ermöglichen. Ausnahmen kann es bei sehr grossen Datenvolumen geben, dazu zählen z.B. Luftbilder oder Laserscanndaten. Daten welche auf dem Netzwerk bearbeitet werden sind nicht zu sichern.

In die wöchentliche Sicherung werden alle Daten vom Netzlaufwerk aufgenommen. (mit Ausnahmen von besonders grossen Datensätzen)

- **Welche Speichermedien werden verwendet?**

Die täglichen Sicherungen der lokalen Daten erfolgen auf das Netzwerk in das Gruppeneigene Verzeichnis. Es wird pro Tag ein Verzeichnis angelegt, dieser Verzeichnisbaum wird nach der Wöchentlichen Sicherung gelöscht. Anschliessend werden die Verzeichnisse wieder frisch angelegt.

Die Wöchentlichen Sicherungen werden auf eine CD gebrannt.

- **Wo werden sich diese Speichermedien befinden?**

Tägliche Sicherungen werden auf dem internen Netzwerk durchgeführt, wo sie auch verbleiben bis zu ihrer Löschung.
Die Wöchentlichen Sicherungen verbleiben bei der verantwortlichen Person (ausserhalb der Schule).

- **Wie lange werden diese Datensicherungen aufbewahrt?**

Die Löschung der täglichen Sicherungen der aktuellen Woche erfolgt erst nachdem die Wöchentliche Sicherung überprüft wurde.

Die CDs der Wöchentlichen Sicherungen bleiben bis Diplomarbeitsende erhalten.

- **Qualitätssicherung?**

Es wurde eine Checkliste für die Tages- und Wochensicherung erstellt welche zwischen unseren Arbeitsplätze im Zimmer 632 an der Wand hängt.

B. Lüscher S. Nebiker

…………………….. ……………………….

12.5 Besprechungen

12.5.1 Zwischenbesprechung 1

Protokoll

Datum: Donnerstag, 29. September 2005
Zeit: 10.15 – 11.00 Uhr
Ort: Fachhochschule beider Basel, Zimmer 624
Anwesend: Herr Nebiker, Herr Budmiger, Herr Knabel, , D. Emmenegger
Protokoll: Th. Schuler

Dieses Protokoll beschränkt sich auf die Erklärungen und gefassten Beschlüsse welche aus dem abgegebenen Dokument (Unterlagen zur ersten Zwischenbesprechung) während der Diskussion entstanden sind.

Ablauf der Besprechung
- Überblick zum Ablauf, Vorgehen und Stand der Arbeit, Detailbesprechung zu den einzelnen Projektzielen sowie Beschlussfassungen.

Aufgabenstellung und -Unterteilung
- wurde kurz besprochen und als vernünftig betrachtet

Schwergewichte / Prioritäten
- Einzelne Schritte und Ablauf als gut bezeichnet, allerdings werden Prioritäten durch Reihenfolge der Arbeitschritte (Etappen) bezeichnet, da alle Punkte erreicht werden können.

Stand der Arbeiten
- wurde zur Kenntnis genommen

Weiteres Vorgehen

Grundlagen
- Vegetation
  - Bisherige Baumvisualisierung auch behandeln(Symbole, „Kreuzdarstellung“). Es wurde auf die Dissertation von Herrn Häberling verwiesen.
  - Untersuchungen wie man schnell und einfach aus Laserscanning-Daten Bäume modellieren kann; Möglichkeiten:
    - nur Höhe/Breite -Informationen verwenden
    - Bäume selber modellieren (Stichworte: Partikel-Konstruktion)
    - Billboard-Bäume“ welche sich in der Szene drehen
    - Erstellen eines Datenflusses

Animation
- es wurde das Buch „Digital Audio / Video“ ausgehändigt
- Beschränkung der Animationsmöglichkeiten auf Fahnen/Feuer/Rauch/Springbrunnen/Mobiliar/Türen welche auf und zu gehen
- stehende 3D-Leute in einer Szene mit kleinen Bewegungen werden erwünscht, angestrebt

Berichtaufbau

- modularer Aufbau welcher zu den einzelnen Teilgebieten folgende Untertitel beinhaltet: Grundlagen(Theorie) / Untersuchungen / Resultate / Diskussion – Was heisst, dass zum Beispiel bei der bereits existierende Unterteilung zur Texturierung alles unter Grundlagen fallen wird.

Arbeitsplan

- etwas zu viel Zeit für Poster / Internet einkalkuliert
- Realisierungszeit besser hervorheben
- genügend Reserven einplanen
- Vergleich „MicroStation/Cinema 4D“ erst nach „Visualisierung Internet“

=> wurden geändert

Nächste Besprechung:
Datum: Montag, 24. Oktober 2005
Zeit: 10.30 – 12.00 Uhr
Ort: Fachhochschule beider Basel, Zimmer 624

Falls wir von ihnen nichts mehr hören, gilt das Protokoll als genehmigt.
Vielen Dank für ihre Kenntnisnahme

freundliche Grüsse

das Projektteam
Th. Schuler D. Emmenegger

12.5.2 Zwischenbesprechung 2

Protokoll

Datum: Montag, 24. Oktober 2005
Zeit: 10.10 – 11.45 Uhr
Ort: Fachhochschule beider Basel, Zimmer 632
Anwesend: Herr Nebiker, Herr Budmiger, Herr Knabel, Th. Schuler, D. Emmenegger
Protokoll: Th. Schuler

Cinema4D

Texturierung

- Es wurde ein Tool zur Verzeichnungskorrigierung vorgestellt (PTLens), welches allerdings für Messverfahren als ungeeignet betrachtet wurde. Deshalb ist der PHIDAS Weg weiter zu verfolgen.
- Die Möglichkeit beim VRML-Export die Einzelnen Texturen zu resamplen ist wahrscheinlich die Einzige Möglichkeit einer LoD-Verwirklichung.
- zur radiometrischen Korrektur wurde auf Photoshop und Paintshop verwiesen.

Vegetation

- Die gemachten Untersuchungen wurden zur Kenntnis genommen
- Von Herr Nebiker wurde auf einen Artikel verwiesen, welcher sich mit der automatischen Klassifizierung von Laserscanningdaten befasst. Der Artikel wird von Herr Nebiker noch nachgereicht.
- Es muss eine Gegenüberstellung der verschiedenen Varianten zur Visualisierung der Vegetation (Symbolbäume / vbexteriors) erstellt werden.
- Eine Untersuchung / Ablauf wie man schnell und einfach von den Laserscannigdaten (bezüglich Lage / Form) zu einer Visualisierung in Cinema 4D über die vbexteriors^{3DPlants} kommen kann fehlt.

→ wird erledigt

- Bei der Grastexturierung wurde keine Entscheidung getroffen. Es wurde erwähnt, dass die Firma vbvisual.com ein Plug-In für Wiesendarstellung hat.

Animationen

- wurde zur Kenntnis genommen

Videothematik

- Film von 3DGI wurde zur Kenntnis genommen.
- Es wird kein Filmkunstwerk erwartet.
- Es werden grosse Erwartungen an die Besprechung vom 1.November mit der HGK Basel gesetzt. Nach diesem Treffen sollten folgende Punkte geklärt sein:
  - Wie das Projekt filmisch umgesetzt werden kann
  - Thema Kameraführung
- Schwerpunkt ist nicht die Technik, sondern Grundregeln einer Visualisation.

Visualisierung im Internet

- Thematik Plug-In von 3DStudioMax Export in G-Vista erfordert Abklärungen durch Herr Th. Knabel
- Wegen Vrml-Export aus Cinema 4D in G-Vista wird an Pascal Imoberdorf der Firma Geonova verwiesen.

Bericht
- Der Bericht ist zentraler Bestandteil der Arbeit. Es ist auf eine saubere und vollständige Dokumentation aller Untersuchungen / Abläufe / Empfehlungen zu achten.
- Provisorisches Inhaltsverzeichnis ist dem Protokoll beizulegen.

Weiteres Vorgehen
bis Woche 43 fertig mit Recherchen und Untersuchungen
- Woche 44 Umsetzung des Projektes Wildenstein
    - sehr Umfangreich
    - Innenraumanimation
    - Aussenanimationen(Feuer, Fahnen, Springbrunnen etc.)
- Woche 45 Umsetzung des Projektes Homburg
    - Evt. Orientierung von Bildern
    - Grundlagen verbessern/anpassen (DGM, Treppen …)
- Woche 46
    - Bericht
    - Poster
    - Reserve für Umsetzung
- Woche 47
    - Fertigstellung
- Wurde zur Kenntnis genommen. es wird nicht mit letzter Konsequenz eine "perfekte" Visualisation (Film) gefordert, da die Dokumentation nicht zu vernachlässigen ist.

Nächste Besprechung (Schlussbesprechung):
Datum: Dienstag, 13. Dezember 2005
Zeit: 10.30 – 12.00 Uhr
Ort: Fachhochschule beider Basel, Zimmer 624

Falls wir von ihnen nichts mehr hören, gilt das Protokoll als genehmigt.
Vielen Dank für ihre Kenntnisnahme

Mit freundliche Grüsse

das Projektteam
Th. Schuler D. Emmenegger